**TUGAS AKHIR – SS141501**

# *ADS FILTERING* MENGGUNAKAN JARINGAN SYARAF TIRUAN *PERCEPTRON, NAÏVE BAYES CLASSIFIER* DAN REGRESI LOGISTIK

Achmad Fachrudin Rachimawan
NRP 1311 100 123

Dosen Pembimbing
Dr. Brodjol Sutijo Suprih Ulama, M.Si

Program Studi S1 Statistika
Fakultas Matematika Dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabava 2016

`

**TUGAS AKHIR – SS141501**

# *ADS FILTERING* MENGGUNAKAN JARINGAN SYARAF TIRUAN *PERCEPTRON, NAÏVE BAYES CLASSIFIER* DAN REGRESI LOGISTIK

Achmad Fachrudin Rachimawan
NRP 1311 100 123

Dosen Pembimbing
Dr. Brodjol Sutijo Suprih Ulama, M.Si

Program Studi S1 Statistika
Fakultas Matematika Dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2016

*(Halaman sengaja dikosongkan)*

**FINAL PROJECT – SS141501**

# *ADS FILTERING* USING NEURAL NETWORK PERCEPTRON, NAÏVE BAYES CLASSIFIER AND LOGISTIC REGRESSION

Achmad Fachrudin Rachimawan
NRP 1311 100 123

Supervisor
Dr. Brodjol Sutijo Suprih Ulama, M.si

Undergraduate Progamme of Statistics
Faculty Of Mathematics and Natural Science
Sepuluh Nopember Institute Of Technology
Surabaya 2016

*(Halaman sengaja dikosongkan)*

# LEMBAR PENGESAHAN

## *ADS FILTERING* MENGGUNAKAN JARINGAN SYARAF TIRUAN PERCEPTRON, NAÏVE BAYES CLASSIFIER DAN REGRESI LOGISTIK

**TUGAS AKHIR**
Diajukan untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Sains
pada
Program Studi S-1 Jurusan Statistika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember

Oleh :
**ACHMAD FACHRUDIN RACHIMAWAN**
**NRP 1311 100 123**

Disetujui oleh Pembimbing Tugas Akhir

**Dr. Brodjol Sutijo Suprih Ulama, M.Si**
**NIP. 19660125 199002 1 001** ( )

Mengetahui
Ketua Jurusan Statistika FMIPA-ITS

**Dr. Suhartono**
**NIP. 19710929 199512 1 001**

SURABAYA, JANUARI 2016

`

**_Ads Filtering_ Menggunakan Jaringan Syaraf Tiruan _Perceptron_, _Naïve Bayes Classifier_ dan Regresi Logistik**


Nama : Achmad Fachrudin Rachimawan
NRP : 1311100123
Jurusan : Statistika
Pembimbing : Dr. Brodjol Sutijo Suprih Ulama, M.Si


## ABSTRAK


*Email merupakan fasilitas yang mutlak diperlukan dalam berbagai bidang. Pentingnya email dan jumlahnya yang begitu banyak menyebabkan penyalahgunaan. Salah satu penyalahgunaan yang sering ditemui adalah email iklan yang dikirimkan oleh perusahaan penyedia konten internet saat pengguna mendaftar pada situs perusahaan tersebut. Terdapat metode agar email iklan dari perusahaan-perusahaan tersebut bisa secara otomatis dikenali yaitu klasifikasi. Data email berbentuk teks, sehingga jauh lebih rumit dan perlu proses untuk mempersiapkan data. Salah satu prosesnya adalah pembobotan ads atau adicity. Metode klasifikasi yang digunakan adalah Naive Bayes Classifier (NBC) yang secara umum sering digunakan dalam data teks dan Perceptron yang diketahui merupakan metode yang cukup sederhana untuk menyelesaikan permasalahan kompleks. Kedua metode tersebut akan dibandingkan dengan metode regresi logistik untuk mengetahui akurasi paling baik. Hasil penelitian menunjukkan bahwa NBC lebih unggul dibanding Perceptron* dan Regresi Logistik, *dan pada NBC False Positive Ratio lebih mudah untuk dikontrol.*



*Kata Kunci : email, iklan, klasifikasi, naïve bayes, perceptron*

`

***Ads Filtering Using***
***Neural Network Perceptron,***
***Naïve Bayes Classifier, and Logistic Regression***


| | |
|---|---|
| Name | : Achmad Fachrudin Rachimawan |
| NRP | : 1311100123 |
| Department | : Statistics |
| Supervisor | : Dr. Brodjol Sutijo Suprih Ulama, M.Si |


## ABSTRACT


*Email is a facility that is absolutely necessary in various fields. The importance of email its huge numbers causes of abuse. One of them is the advertisement emails sent by the provider of Internet content when users register on their company's website. There are several methods to solve that problem, one of them that advertisement email from these companies can automatically recognize is classification. Data on email is a text form, so it's much more complicated and need a different process to prepare the data. One of the process is the weighting of ads or adicity. The classification method used in this research is Naive Bayes classifier (NBC), which is often used in text data and Perceptron that known both of them which are fairly simple method to solve complex problems. Both of these methods will be compared with logistic regression to determine the best results. The results showed that the NBC superior to Perceptron and Logistic Regression , and on NBC False Positive Ratio is easier to control.*



*Key Words : email, ads, classification, naïve bayes, perceptron.*

*(Halaman sengaja dikosongkan)*

*(Halaman sengaja dikosongkan)*

`

# KATA PENGANTAR

Puji syukur kepada Allah SWT karena dengan rahmat, ridho, serta bantuan-Nya tugas akhir yang berjudul "Ads Filtering Menggunakan Jaringan Syaraf Tiruan Perceptron dan Naïve Bayes Classifier" dapat terselesaikan sebagai salah satu syarat kelulusan di Jurusan Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Sepuluh Nopember Surabaya. Selesainya tugas akhir ini, menandakan selesai pula masa studi yang telah ditempuh selama empat tahun setengah. Diharapkan tugas akhir ini dapat memberikan manfaat bagi pembaca, juga menjadi salah satu batu loncatan bagi untuk terus berkarya, dan memberikan sumbangsih bagi ilmu pengetahuan. Sadar bahwa dalam proses pengerjaan hingga terselesaikanya tugas akhir ini dibutuhkan bantuan dari berbagai pihak dan diucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang membantu pengerjaan tugas akhir ini, antara lain:

- Orang tua, dan keluarga besar yang senantiasa memberikan dukungan penuh dan telah berkorban banyak. Terimakasih atas doa dan dorongan untuk mencapai yang terbaik.
- Seluruh Bapak dan Ibu dosen yang telah memberikan ilmunya selama berkuliah disini. Semoga ilmu tersebut selalu bermanfaat.
- Bapak Dr. Brodjol Sutijo Suprih Ulama, M.Si selaku dosen pembimbing. Terimakasih atas waktu dan ilmu yang sudah diberikan selama proses pembuatan tugas akhir ini, sungguh luar biasa.
- Bapak Ir. Dwiatmono Agus W M.Ikom dan Bu Dr. Kartika Fitriasari M.Si selaku dosen penguji. Terima kasih atas waktunya dan ilmu juga saran terkait pengerjaan tugas akhir ini.

- Bapak Heri Kuswanto dan Pak Kresna terimakasih atas ilmu dan pengalamannya yang selalu memotivasi.
- Teman-teman Statistika 2011, senior, dan junior yang telah menemani selama berada di ITS. Terimakasih atas waktu yang telah diberikan.
- Yudha, Ferdi terima kasih waktu ngobrol dan minum kopinya.
- Fakhrul, Dika, Irsyad, Indi, Irma dan kawan lainnya sebagai tempat diskusi dan *curhat*.
- Teguh, Sistem Informasi Sistem Informasi UPN terimakasih atas waktu yang diberikan untuk dapat menyelesaikan tugas akhir ini.
- Kiki terima kasih sudah banyak menemani dan menyemangati.
- Civitas akademi lainnya yang telah membantu proses perkuliahan disini.

Dan berbagai pihak yang tidak dapat dituliskan namanya satu per satu. Semoga Tuhan membalas semua kebaikan yang telah dilakukan. Dari banyaknya kekurangan yang dapat dikembangkan pada tugas akhir ini, oleh karena itu akan diterima saran dan kritik yang membangun. Semoga tugas akhir ini dapat memberikan manfaat.

Surabaya, Januari 2016

Penulis

`

# DAFTAR ISI

Halaman

`

# DAFTAR GAMBAR

*(Halaman sengaja dikosongkan)*

`

# DAFTAR TABEL

*(Halaman sengaja dikosongkan)*

`

## DAFTAR LAMPIRAN

*(Halaman sengaja dikosongkan)*

# BAB I
# PENDAHULUAN

## 1.1 Latar Belakang

Perkembangan teknologi saat ini telah tumbuh dengan luar biasa pesat, khususnya di bidang teknologi komunikasi dan informasi. Dengan keberadaan internet, segala informasi dan berita dapat diterima dan diakses oleh setiap orang. Bahkan dengan internet, setiap orang dapat mengirim dan menerima pesan dari satu orang ke orang lainnya dengan mudah menggunakan sebuah pesan elektronik, ataupun dengan mengunakan media sosial. Pesan elektronik yang lebih dikenal sebagai email merupakan fasilitas yang saat ini menjadi sarana yang mutlak diperlukan dalam berbagai bidang, mulai dari bidang industri, pendidikan, kesehatan, dll. Tetapi tidak semua orang menggunakan email dengan baik dan benar, bahkan dapat menyebabkan kerugian bagi orang lain. Hal ini dikarenakan fasilitas email yang murah dan mudah digunakan oleh setiap orang, sehingga mengakibatkan banyaknya penyalahgunaan pada penggunaan email itu sendiri, atau yang biasa disebut dengan email spam atau *bulkmail* yang biasanya berisi beragam tujuan, diantaranya adalah penipuan (berkedok amal, undian lottere), pencucian uang atau *money loundring* (menawarkan transaksi pekerjaan yang berhubungan dengan transaksi bank), atau bahkan menyebarkan virus. (Suyanto, 2014).

Namun pada saat ini para pengguna fasilitas email sudah tidak perlu khawatir karena penyedia server email telah menggunakan beragam metode dan teknik spam filtering, misalnya *Real Time Block List* yang membandingkan alamat IP pengirim email dengan IP *Global Blacklist*, yang dengan begitu email spam akan mudah untuk dikenali (Pivotal, 2007). Permasalahan kemudian justru muncul dari penggunaan akun email pengguna yang aktif men-*download* berbagai macam konten dari beragam situs web atau perusahaan penyedia konten.

Dengan melakukan *register* atau pendaftaran di situs atau perusahaan penyedia konten, para pengguna internet memiliki keuntungan yaitu diperbolehkan untuk mendownload beragam konten yang disediakan secara gratis, namun di sisi lain pemilik situs web atau perusahaan penyedia konten bisa dengan leluasa mengirimkan email yang berisi iklan atau promosi produk yang biasa disebut *Ads* atau *Advestisement*. *Ads* tidaklah berbahaya, dibandingkan dengan spam, *Ads* tidak mengandung unsur penipuan ataupun virus, hanya saja *Ads* dirasa sudah sangat *annoying* atau mengganggu (Liliweri, 2011). Pada bulan Juni 2005, jumlah email yang berisi iklan sudah mencapai 30 milyar perhari, dan setahun kemudian pada Juni 2006 email yang berisi *Ads* atau iklan meningkat jumlahnya menjadi hampir 65 milyar perhari, dan terakhir meningkat menjadi 90 milyar email perhari pada tahun 2007 (IronPort, 2007). Pada studi yang dilakukan *Messaging Anti-Abuse Working Group* tahun 2005, menyatakan bahwa banyak perusahaan di Amerika Serikat mengalami kerugian akibat produktivitas pegawai menurun akibat waktu yang dikeluarkan untuk menghapus email yang berisi *Ads* atau iklan (MAAWG, 2005).

*Ads* umumnya tidak dikenali oleh teknik spam *blocking* karena IP para pengirim *Ads* bukanlah termasuk ke dalam *Global Balcklist* karena pengirim *Ads* adalah perusahaan yang secara resmi terdaftar sebagai perusahaan penyedia konten. Melakukan *self blocking* terhadap alamat IP perusahaan penyedia konten juga akan merugikan pengguna karena tidak semua email yang dikirim oleh perusahaan penyedia konten adalah *Ads*, terkadang email yang dikirim merupakan informasi penting yang harus segera ditindak lanjuti oleh pengguna.

Ada beragam metode dalam teknik *learning* (machine) yang dapat digunakan untuk memilah *Ads* seperti *Random Forest*, SVM (Support Vector Machine), KNN (K Nearest Neighbors), dan masih banyak lagi. Dan dari beragam teknik tersebut yang digunakan dalam penelitian ini Adalah jaringan syaraf tiruan *Perceptron*, *Naive Bayes Classifier* dan Regresi Logistik.

Digunakannya ketiga metode tersebut karena secara umum sering digunakan dalam data teks, keduanya juga merupakan metode yang cukup sederhana untuk menyelesaikan permasalahan yang kompleks. Suatu email bisa berisi sekitar 4000 kata, sementara email yang lain hanya berisi 3 kata. Jika dianggap bahwa setiap kata sebagai atribut atau dimensi, maka setiap email bisa memiliki dimensi yang berbeda dengan email lainnya (Suyanto, 2014). Padahal, jaringan syaraf tiruan *perceptron* membutuhkan dimensi masukan yang sama. Maka *preprocessing data* dibutuhkan guna memenuhi kebutuhan dari metode yang digunakan (Moch Agus, 2007).

Menurut Suyanto (2014), terdapat 2 jenis proses pembelajaran pada jaringan syaraf tiruan, yaitu pembelajaran dengan pengawasan *(supervised learning)* dan pembelajaran tanpa pengawasan *(unsupervised learning).* Proses pembelajaran pada jaringan syaraf disebut dengan pengawasan bila *output* yang diharapkan sudah diketahui sebelumnya atau telah ditentukan target, sehingga proses pembelajaran berjalan dengan terarah, proses ini mirip dengan motode klasifikasi yang sebelum dilakukan proses klasifikasi, target atau *output* yang diharapkan telah ditentukan. Pada metode pembelajaran tanpa pengawasan, tidak memerlukan target *output* sehingga proses berlangsung tak terarah, proses ini mirip dengan proses *clustering*. Untuk melakukan pengenalan *email*, pembelajaran terawasi lebih cocok karena menggunakan target keluaran. Diantaranya yang termasuk metode pembelajaran terawasi adalah *Delta Rule*. *Delta rule* merupakan suatu metode pembelajaran pada jaringan syaraf tiruan yang digunakan untuk melakukan *update* bobot yang didasarkan pada nilai delta (selisih) antara nilai keluaran (output) dan nilai target. Sedangkan jenis jaringan yang memiliki arsitektur sederhana yang sering digunakan adalah *Perceptron. Perceptron* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemapuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang

`

serupa (tetapi tidak sama) dengan pola yang dipakai selama pelatihan (Siang, 2005). Penelitian oleh Owen & Richard (2012) dengan menggunakan jaringan syaraf tiruan *perceptron* untuk klasifikasi email spam menghasilkan 5,02% error pada iterasi sebanyak 1000 kali.

Metode kedua yang digunakan dalam penelitian ini adalah *Naïve Bayes Classifier* (NBC*)*, NBC telah banyak digunakan dalam penelitian mengenai *text mining* dan *spam filtering*, beberapa kelebihan NBC diantaranya adalah sederhana tapi memiliki akurasi yang tinggi. (Miller, 2005) Penelitian berkaitan dengan metode NBC telah dilakukan diantaranya oleh Durajati, C & Gumelar, A, B (2012) menggunakan NBC, menghasilkan ketepatan klasifikasi sebesar 87% dan menyimpulkan bahwa semakin banyak data training semakin baik. Anugroho, P (2012) menggunakan *naïve bayes classifier* untuk mengklasifikasikan email spam menghasilkan tingkat error sebesar 4,83%, dan juga menyimpulkan bahwa *naïve bayes classifier* mempunyai tingkat error yang besar jika terdapat selisih pada jumlah keyword yang ada di data training. Lestari, Putra, & Cahyawan (2013), melakukan klasifikasi tipe kepribadian orang dengan menggunakan NBC dan menghasilkan ketepatan klasifikasi sebesar 92,5%. Dalam penelitian ini, kedua metode non-parametrik (Naïve Bayes dan Perceptron) akan dibandingkan dengan metode parametrik yaitu Regresi Logistik dan dicari metode yang menghasilkan tingkat galat paling kecil.

## 1.2 Rumusan Masalah

Klasifikasi teks umumnya menggunakan metode NBC karena dirasa sederhana dan mudah untuk diterapkan, dalam penelitian ini metode NBC akan dibandingkan tingkat akurasinya dengan jaringan syaraf tiruan yang mempunyai arsitektur yang juga sederhana yaitu perceptron dan metode parametrik yaitu regresi logistik. Dengan demikian, pada penelitian kali ini dapat dirumuskan suatu permasalahan yaitu bagaimana perbandingan

tingkat akurasi antara metode NBC, jaringan syaraf tiruan perceptron dan regresi logistik?

## 1.3 Tujuan

Tujuan penulisan skripsi ini adalah sebagai berikut.

1. Mengetahui hasil proses klasifikasi *Ad atau Ad filtering* pada email menggunakan jaringan syaraf tiruan *perceptron*, naive *bayes classifier* (NBC), dan Regresi Logistik.
2. Mengetahui perbandingan tingkat akurasi antara metode NBC, Jaringan Syaraf Tiruan Perceptron, dan Regresi Logistik.

## 1.4 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat bermanfaat dalam bidang klasifikasi email yang mengandung iklan atau *Ads* secara umum dengan menggunakan metode NBC, *perceptron* dan regresi logistik. Penelitian ini juga diharapkan dapat membantu pengguna email untuk secara otomatis menolak email yang berisi iklan atau *Ads*.

## 1.5 Batasan Masalah

Dalam penelitian ini terdapat beberapa batasan yang digunakan sebagai berikut.

a. Data yang digunakan merupakan email berbahasa inggris berjumlah 5056 yang merupakan inbox dari sebuah perusahaan selama beberapa kurun waktu, email didapatkan dari http://spamassassin.apache.org/publiccorpus/.
b. Email dikategorikan menjadi 2, yaitu email yang mengandung *Ad* dan email yang tidak mengandung *Ads*.
c. Pada jaringan syaraf tiruan dengan arsitektur perceptron, metode pembelajaran yang digunakan adalah *delta rule.*

`

*(Halaman sengaja dikosongkan)*

# BAB II
# TINJAUAN PUSTAKA

## 2.1 *Advertisement*

Pengertian *Ad* atau iklan menurut bahasa adalah memperkenalkan suatu barang, produk dan mempromosikan barang atau pun jasa baik secara online mau pun offline yang disampaikan melalui media dan di biaya oleh pemrakarsa yang dikenal serta di tunjuk sebagian masyarakat melalui ,radio, televisi, surat kabar, majalah dan lain-lain. Dan iklan juga di definisikan sebagai pesan yang menawarkan produk yang ditujukan kepada masyarakat lewat suatu media masa (kasali, 1995). Menurut Liliweri (2011) contoh iklan juga merupakan salah satu faktor bentuk komunikasi bertujuan untuk memersuasi para pendengar dan pembaca agar mereka memutuskan untuk melakukan tindakan tertentu. Sedangkan *Non-Ad* adalah segala sesuatu yang tujuannya bukan untuk komersialisasi atau bukan untuk memersuasi pembaca agar melakukan sesuatu yang diinginkan oleh penulis (kasali, 1995).

## 2.2 *Text Mining*

Istilah *data mining* adalah mencari pola dalam data. Demikian pula dengan *text mining* tentang mencari pola dalam teks. *Text mining* adalah proses menganalisis teks untuk mengekstrak informasi yang berguna untuk tujuan tertentu. Apabila dibandingkan dengan jenis data yang lain, sifat data berbentuk teks tidak terstruktur dan sulit untuk menangani. Diharapkan melalui proses *text mining*, informasi yang ada dapat diekstraksi secara jelas di dalam teks tersebut dan dapat dipergunakan dalam proses analisis menggunakan alat bantu komputer. Tujuan dari text mining adalah untuk mendapatkan informasi yang berguna dari sekumpulan dokumen. Jadi, sumber data yang digunakan pada text mining adalah kumpulan teks yang memiliki format yang tidak terstruktur atau minimal semi

terstruktur. Adapun tugas khusus dari text mining antara lain yaitu pengkategorisasian teks (text categorization) dan pengelompokan teks text clustering). (Witten et al, 2011)

## 2.3 Klasifikasi Text

Klasifikasi teks merupakan proses menemukan pola baru yang belum terungkap sebelumnya. Klasifikasi teks dilakukan dengan memproses dan menganalisa data dalam jumlah besar. Dalam prosesnya, klasifikasi teks melibatkan struktur yang mungkin terdapat pada teks dan mengekstraks informasi yang relevan pada teks. Dalam menganalisis sebagian atau keseluruhan teks yang tidak terstruktur, klasifikasi teks mencoba mengasosiasikan sebagian atau keseluruhan satu bagian teks dengan yang lainnya berdasarkan aturan-aturan tertentu. (Miller, 2005).

## 2.4 Praproses Text

Tahapan praproses ini dilakukan agar dalam klasifikasi dapat diproses dengan baik. Tahapan dalam praproses teks adalah sebagai berikut:

a. *Case Folding*, merupakan proses untuk mengubah semua karakter pada teks menjadi huruf kecil. Karakter yang diproses hanya huruf ‘a’ hingga ‘z’ dan selain karakter tersebut akan dihilangkan seperti tanda baca titik (.), koma (,), dan angka (Weiss, 2010).
b. *Tokenizing*, merupakan proses memecah yang semula berupa kalimat menjadi kata-kata atau memutus urutan string menjadi potongan-potongan seperti kata-kata berdasarkan tiap kata yang menyusunnya (Ahmad, 2004).
c. *Stopwords*, merupakan kosakata yang bukan merupakan kata unik atau ciri pada suatu dokumen atau tidak menyampaikan pesan apapun secara signifikan pada teks atau kalimat. (Dragut, Fang, Sistla, Yu, & Meng, 2009). Kosakata yang dimaksudkan adalah kata penghubung dan

kata keterangan yang bukan merupakan kata unik misalnya “sebuah”, “oleh”, “pada”, dan sebagainya.

d. *Stemming*, yakni proses untuk mendapatkan kata dasar dengan cara menghilangkan awalan, akhiran, sisipan, dan *confixes* (kombinasi dari awalan dan akhiran).

## 2.5 *Information Gain*

Karena pada setiap penelitian *text mining* melibatkan data atau informasi dari sumber data yang berbeda-beda, maka metode penanganan data dalam tahapan praproses pun menjadi berbeda-beda, sehingga metode yang akan digunakan diketahui setelah dilakukan praproses (Moch Agus, 2007). *Information gain* bisa dianggap masuk ke dalam praproses teks, digunakan untuk mengukur efektivitas suatu atribut dalam mengklasifikasikan data. Tahapan dalam *Information Gain* adalah dengan membuat :

a. *Wordlist* atau *dictionary*, dari seluruh jumlah email (5056) dikodekan menjadi angka-angka, dengan cara dibuat daftar seluruh kata yang pernah muncul pada email tersebut. Misal dari seluruh email ditemukan terdapat 200.000 kata, maka setiap kata dalam tiap email akan diganti dengan menggunakan angka-angka berdasar jumlah angka yang ditemukan.
b. *Feature Vectors Content,* Memilih (misal) sebanyak 100 kata dari keseluruhan kata yang terdapat pada email. Tiap kata mengadung sebuah fitur yang di dalamnya terkandung bobot untuk tiap-tiap kata.
c. *Features (Words) Selection*, setelah diambil ketentuan tentang banyaknya *Vectors Content* yang diambil, maka selanjutkan akan dipilih kata-kata yang mewakili atau merepresentasikan email dengan bobot tertinggi yang kemudian dipersiapkan untuk *Vectors Content*. Dengan langkah,
   1. Menghitung Bobot Ad (*Ad-icity)*
      **Ad-icity** bisa ditulis:

$$Ad-icity(w) = \frac{\Pr(w \mid A)}{\Pr(w \mid A) + \Pr(w \mid \mathrm{B})} \qquad (2.1)$$

`

Dengan:

Pr(w|A) = peluang bahwa sebuah kata muncul di email non iklan.

Pr(w|B) = peluang bahwa sebuah kata muncul di email iklan.

Tiap peluang dihitung berdasarkan proporsi yang relevan dalam data training.

2. *Selection of Word*, dalam proses ini dilakukan *rangking* untuk tiap kata, dan (misal dengan jumlah 100) kata dengan rank tertinggi kemudian dipilih. Praktek intuitifnya adalah untuk menetapkan peringkat tinggi untuk kata-kata yang *Ad-icity* yang jauh dari 0,5 (tinggi atau lebih rendah). Jika Ad-icity mendekati nilai 1, maka dapat dikatakan kata tersebut adalah indikator *Ads* yang baik, dan sebaliknya jika mendekati 0 maka dapat dikatakan kata tersebut adalah indikator normal yang baik.

   Namun akan ditemukan, bahwa hanya melihat dari *Adicity* tidaklah cukup, kata dengan kedua Peluang Pr(w|A) dan Pr(w|B) nya terlalu kecil tidak bisa dijadikan sebuah indikator yang baik. Walaupun *Ad-icity* kata tersebut jauh dari 0.5 (mendekati 0 atau 1).

   Maka perlu dilihat pula seberapa besar selisih absolut seperti berikut :

   $$|Pr(w|A) - Pr(w|B)|$$

   Proses pemilihan kata akan berlangsung :

   - Saring atau temukan kata dengan $|Ad\text{-}icity - 0.5| < 0.05$
   - Saring atau temukan kata yang jarang, yaitu yang muncul di kedua kategori yang kurang dari threshold yang diberikan (1%).
   - Untuk tiap kata yang tidak tersaring, hitung |Pr(w|A) - Pr(w|B)|
   - Pilih kata dengan |Pr(w|A) - Pr(w|B)| terbesar.

d. Shuffling (pengacakan)

Rasio Ad-normal berubah-ubah (dinamis) dari waktu ke waktu, dan itu adalah suatu hal yang menarik, dari masalah ini ditemukan bahwa Naïve Bayes classifier menyesuaikan dengan cepat untuk rasio baru yang dilatih secara bertahap. Maka data yang digunakan diambil secara acak baik pada pelatihan dan uji set sebelum digunakan, sehingga rasio Ad-normal dari waktu ke waktu dapat dipelajari oleh algoritma yang telah dibentuk. (Suyanto, 2014)

## 2.6 Jaringan Syaraf Tiruan (Artificial Neural Network)

Jaringan syaraf merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran.

Jaringan Syaraf Tiruan (JST) adalah sistem pemrosesan informasi yang memiliki karakteristik unjuk kerja tertentu yang menyerupai jaringan syaraf biologis (Fausett, 1994). JST telah dikembangkan sebagai generalisasi model matematika dari aspek kognitif manusia atau syaraf biologi, yaitu didasarkan pada asumsi-asumsi bahwa:

a. Pemrosesan informasi terjadi pada elemen-elemen yang disebut *neuron*;
b. Sinyal-sinyal merambat di antara neuron melalui interkoneksi;
c. Setiap interkoneksi memiliki bobot yang bersesuaian pada kebanyakan jaringan syaraf berfungsi untuk mengalikan sinyal yang dikirim;
d. Setiap *neuron* menerapkan fungsi aktivasi pada masukan jaringan untuk menentukan sinyal keluaran.

`

### 2.6.1 *Perceptron*

Perceptron juga termasuk salah satu bentuk jaringan syaraf yang sederhana. Perceptron biasanya digunakan untuk mengklasifikasikan suatu tipe pola tertentu yang sering dikenal dengan pemisahan secara linear. Pada dasarnya, perceptron pada jaringan syaraf dengan satu lapisan memiliki bobot yang bisa diatur dan suatu nilai ambang (*threshold*). Algoritma yang digunakan oleh aturan perceptron ini akan mengatur parameter-parameter bebasnya melalui proses pembelajaran. Nilai *threshold* (θ) pada fungsi aktivasi adalah non negative. Fungsi aktivasi ini dibuat sedemikian rupa sehingga terjadi pembatasan antara daerah positif dan daerah negative.

Garis pemisah antara daerah positif dan daerah nol memiliki pertidaksamaan :

$$w_1x_1 + w_2x_2 + b > \theta \qquad (2.2)$$

dengan w adalah bobot dan x adalah input jaringan. Algortma *Perceptron* sama dengan *delta rule* (pada sub bab 2.6.2), yang membedakan hanya pada *Perceptron* garis pemisah menggunkan *threshold* atau suatu nilai ambang batas, jika pada *delta rule* garis pemisah menggunakan nilai 0.

### 2.6.2 *Delta Rule*

Pada delta rule akan mengubah bobot yang menghubungkan antara jaringan input ke output (y_in) dengan nilai target (t). Hal ini untuk dilakukan untuk meminimalkan error selama pelatihan pola. Delta rule untuk memperbaiki bobot ke-I (untuk setiap pola) adalah :

$$\Delta w_i = \alpha\ (t - y\_in) * x_i \qquad (2.3)$$

dengan :

- x = vektor input
- y_in = input jaringan ke input output Y

$$y\_in = \sum_{i=1}^{n} x_i * w_i \qquad (2.4)$$

- t = target (output)
- w = bobot

Nilai w baru diperoleh dari nilai w lama ditambah dengan Δw,

$$w_i = w_i + \Delta w_i \quad (2.5)$$

## 2.7 Naïve Bayes Classifier

Teorema Bayes merupakan teorema yang mengacu konsep probabilitas bersyarat (Tan et al, 2006). Secara umum teorema Bayes dapat dinotasikan pada persamaan berikut:

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)} \quad (2.6)$$

Salah satu metode klasifikasi yang dapat digunakan adalah metode *naive bayes* yang sering disebut sebagai *naive bayes classification* (NBC). Kelebihan NBC adalah sederhana tetapi memiliki akurasi yang tinggi. Dalam algoritma NBC setiap dokumen direpresentasikan dengan pasangan atribut "$a_1$, $a_2$, $a_3$,..., $a_n$" dimana $a_1$ adalah kata pertama, $a_2$ adalah kata kedua dan seterusnya. Sedangkan V adalah himpunan kategori text. Pada saat klasifikasi algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan ($V_{MAP}$). Adapun persamaan $V_{MAP}$ adalah sebagai berikut:

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \; P(v_j \mid a_1, a_2, \ldots, a_n) \quad (2.7)$$

Dengan menggunakan teorema Bayes, maka persamaan (2.8) dapat ditulis menjadi,

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}} \; \frac{P(a_1, a_2, \ldots, a_n \mid v_j) P(v_j)}{P(a_1, a_2, \ldots, a_n)} \quad (2.8)$$

`

Karena nilai P($a_1$, $a_2$, $a_3$,... $a_n$) untuk semua $v_j$ besarnya sama maka nilainya dapat diabaikan, sehingga persamaan (2.9) menjadi:

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}}\ P\left(a_1, a_2, \ldots, a_n \mid v_j\right) P\left(v_j\right) \qquad (2.9)$$

Naive bayes classifier menyederhanakan hal ini dengan mengasumsikan bahwa didalam setiap kategori, setiap atribut bebas bersyarat satu sama lain (Tan et al, 2006). Dengan kata lain persamaan (2.10) dapat dituliskan sebagai berikut:

$$P\left(a_1, a_2, \ldots, a_n \mid v_j\right) = \prod_i P\left(a_i \mid v_j\right) \qquad (2.10)$$

Kemudian apabila persamaan (0.11) disubtitusikan ke persamaan (0.10), maka akan menghasilkan

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}}\ P\left(v_j\right) \prod_i P\left(a_i \mid v_j\right) \qquad (2.11)$$

P($v_j$) dihitung pada saat *training*. Nilainya didapat dengan:

$$P\left(v_j\right) = \frac{\left|\text{text } j\right|}{\left|training\right|} \qquad (2.12)$$

Dimana |*text j*| merupakan jumlah text (email) yang memiliki kategori *j* dalam *training*. Sedangkan |*training*| merupakan jumlah text (email) dalam contoh yang digunakan untuk *training*. Untuk probabilitas kata $a_i$ untuk setiap kategori P($a_i|v_j$), dihitung pada saat *training*. Dimana,

$$P\left(a_i \mid v_j\right) = \frac{\left|n_i + 1\right|}{\left|n + kosakata\right|} \qquad (2.13)$$

Dimana $n_i$ adalah jumlah kemunculan kata $a_i$ dalam dokumen yang berkategori $v_j$, sedangkan $n$ adalah banyaknya seluruh kata dalam dokumen dengan kategori $v_j$ dan $|kosakata|$ adalah banyaknya kata yang terambil dalam contoh pelatihan.

## 2.8 Regresi Logistik

Regresi logistik adalah salah satu model untuk menduga hubungan antara peubah respon kategori dengan satu atau lebih peubah prediktor yang kontinyu ataupun kategori. Peubah respon yang terdiri dari dua kategori yaitu "ya (sukses)" dan "tidak (gagal)", dan dinotasikan 1="sukses" dan 0="gagal", maka akan mengikuti sebaran Bernoulli. Agresti (2002), menyatakan model regresi logistik :

$$\pi(X_i) = \frac{\exp(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + ... + \beta_p X_{ip})}{1 + \exp(\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + ... + \beta_p X_{ip})} \qquad (2.14)$$

Dengan $\pi(x)$ adalah peluang kejadian sukses dengan nilai probabilita $0 \leq \pi(x) \leq 1$ dan $\beta j$ adalah nilai parameter dengan j = 1,2,.....,p. $\pi(x)$ merupakan fungsi yang non linier, sehingga perlu dilakukan transformasi ke dalam bentuk logit untuk memperoleh fungsi yang linier agar dapat dilihat hubungan antara variabel bebas dan variabel tidak bebas. Dengan melakukan transformasi dari logit $\pi(x)$, maka didapat persamaan yang lebih sederhana. Proses pendugaan parameter dari regresi logistik menggunakan metode MLE. Menurut Agresti (2002), metode MLE memberikan nilai duga bagi $\beta$ dengan cara memaksimumkan fungsi likelihood dan mensyaratkan bahwa data mengikuti sebaran Bernoulli. Fungsi likelihood untuk model regresi logistik dikotomus adalah:

$$\xi(\beta) = \prod_{i=1}^{n} f(\beta, y) = \prod_{i=1}^{n} \pi(x_i)^{y_i} (1 - \pi(x_i))^{1-y_i} \qquad (2.15)$$

`

Agar nilai fungsi mencapai maksimum maka turunan parsial pertama terhadap disamadengankan nol. Persamaan hasil turunan masih nonlinier, maka dibutuhkan metode iterasi Newton-Raphson (Agresti, 2002). Pengujian signifikansi parameter model regresi logistik dilakukan secara simultan dan secara parsial. Pengujian secara simultan dilandaskan pada hipotesis:

$H_0 : \beta_1 = \beta_2 = ... = \beta_j = 0$ (tidak ada pengaruh antara peubah prediktor terhadap peubah respon)

$H_1$ : paling sedikit ada satu $\beta_j \neq 0$ (ada pengaruh antara peubah prediktor terhadap peubah respon)

dengan statistik uji G adalah:

$$-2\ln\left[\frac{L_0}{L_1}\right] \sim X^2(p) \qquad (2.16)$$

Dengan $X^2$ adalah derajat bebas yang nilainya sama dengan banyaknya parameter , di mana $H_0$ akan ditolak jika nilai statistik uji G ≥ dengan tingkat kepercayaan (1-α)100. Sedangkan pengujian secara parsial dilandaskan pada hipotesis:

H0 : $\beta_j = 0$ (tidak ada pengaruh antara masing-masing peubah prediktor terhadap peubah respon)

H1 : $\beta_j \neq 0$ (ada pengaruh antara masing-masing peubah prediktor terhadap peubah respon)

Rumus statistik uji *Wald* adalah :

$$\left[\frac{\beta}{Se(\beta_j)}\right] \sim Z \quad ; j = 0,1,2, .....,p \qquad (2.17)$$

Hipotesis nol ditolak jika |W| > Zα/2 artinya peubah prediktor berpengaruh nyata terhadap peubah respon (Hosmer dan Stanley, 2000).

Hosmer dan Stanley (2000) menyatakan bahwa peubah respon dengan dua kategori (biner) dengan ketentuan jika π(x) ≥

0.5 maka hasil prediksi adalah 1, jika $\pi(x) < 0.5$ maka hasil prediksi adalah 0. Klasifikasi menggunakan model peluang dengan persamaan sebagai berikut :

$$\text{logit } \pi(x_i) = \ln\left(\frac{\pi(x_i)}{1-\pi(x_i)}\right) = \beta_0 + \beta_1 X_{i1} + ... + \beta_p X_{ip} \qquad (2.18)$$

Dengan $\pi(x)$ adalah peluang kejadian sukses dan i adalah kategori atau kelas data (email).

## 2.9 Pengukuran Performa

Pengukuran performa dilakukan untuk melihat hasil yang didapatkan dari klasifikasi. Terdapat beberapa cara untuk mengukur performa, beberapa cara yang sering digunakan adalah dengan menghitung akurasi dan *False Positive Ratio*. (Wang, Luo, Wu, & Chi, 2005; Sheu, 2008)

### 2.8.1 Akurasi

Akurasi merupakan persentase dari total dokumen yang teridentifikasi secara tepat dalam proses klasifikasi.

$$akurasi = \frac{jumlah\ klasifikasi\ benar}{jumlah\ dokumen\ uji\ coba} \times 100\% \qquad (2.19)$$

### 2.8.2 *False Positive Ratio*

False positive ratio adalah persetase dari jumlah email *Ad* yang gagal dikenali dibandingkan dengan jumlah email normal.

$$FPR = \frac{\text{jumlah False Positive}}{\text{jumlah email normal (bukan Ads)}} \times 100\% \qquad (2.20)$$

`

*(Halaman sengaja dikosongkan)*

# BAB III
# METODOLOGI PENELITIAN

## 3.1 Sumber Data

Data yang digunakan merupakan email berbahasa inggris berjumlah 5056 yang merupakan inbox dari sebuah perusahaan selama beberapa kurun waktu (3 bulan pada September – Nopember 2009) , email didapatkan dari sebuah lembaga pemerhati email http://spamassassin.apache.org/publiccorpus/.

Email dikategorikan menjadi dua, yaitu email yang mengandung iklan atau *Ads* dan email yang tidak mengandung iklan. Dengan proporsi jumlah email iklan sebanyak 1549 dan email yang tidak mengandung iklan sebanyak 3507. Email diproses menggunakan software R dan sublime Text 2 (Unregistered Version).

## 3.2 Langkah Analisis

1. Menyiapkan data email, membaginya menjadi email berisi iklan dan tidak berisi iklan.
   a) Email yang diambil untuk tahap pelatihan dan pengujian merupakan sampel dari masing-masing kategori email.
   b) Data sampel tersebut dibagi menjadi data *training* dan data *testing* dengan proporsi 50:50, 60:40, 70:30, 80:20, dan 90:10.
2. Praproses Teks

   *Information gain* bisa dianggap masuk ke dalam praproses teks, digunakan untuk mengukur efektivitas suatu atribut dalam mengklasifikasikan data. Tahapan dalam *Information Gain* adalah dengan membuat :

   a. *Wordlist* atau *dictionary*, dari seluruh jumlah email (5000) dikodekan menjadi angka-angka, dengan cara dibuat daftar seluruh kata yang pernah muncul pada email tersebut. Misal dari seluruh email ditemukan terdapat

200.000 kata, maka setiap kata dalam tiap email akan diganti dengan menggunakan angka-angka berdasar jumlah angka yang ditemukan. Dengan kata lain tiap kata akan menjadi variabel predictor untuk respon jenis email.

b. *Feature Vectors Content,* Memilih sebanyak 100 kata dari keseluruhan kata yang terdapat pada email. Tiap kata mengadung sebuah fitur yang di dalamnya terkandung bobot untuk tiap-tiap kata.

c. *Features (Words) Selection*, setelah diambil ketentuan tentang banyaknya *Vectors Content* yang diambil, maka selanjutkan akan dipilih kata-kata yang mewakili atau merepresentasikan email dengan bobot tertinggi yang kemudian dipersiapkan untuk *Vectors Content*.

d. Melakukan pengacakan pada data yang digunakan sebai input baik pada data *training* maupun data *testing* agar klasifikasi semakin baik.

3. Klasifikasi email menggunakan jaringan syaraf tiruan *perceptron.*

   Algoritma pelatihan perceptron adalah sebagai berikut :

   a. Inisialisasi semua bobot dan bias (umumnya wi = $b_0$)

   b. Tentukan laju pemahaman (α). Untuk penyederhanaan biasanya α diberi nilai = 1. Selama ada elemen vector masukan yang respon unit keluarannya tidak sama dengan target, lakukan :

      1) Set aktivasi unit masukan $x_i$ ( i = 1,…n)

      2) Hitung respon unit keluaran : net = $\sum x_i w_i + b$

      3) Perbaiki bobot pola yang mengandung kesalahan ($y \neq t$ )

         Perbaikan bobot pada perceptron dengan mengikuti aturan pada rumus (2.3)

   Ada beberapa hal yang perlu diperhatikan dalam algoritma tersebut :

   a) Iterasi dilakukan terus hingga semua pola memiliki keluaran jaringan yang sama

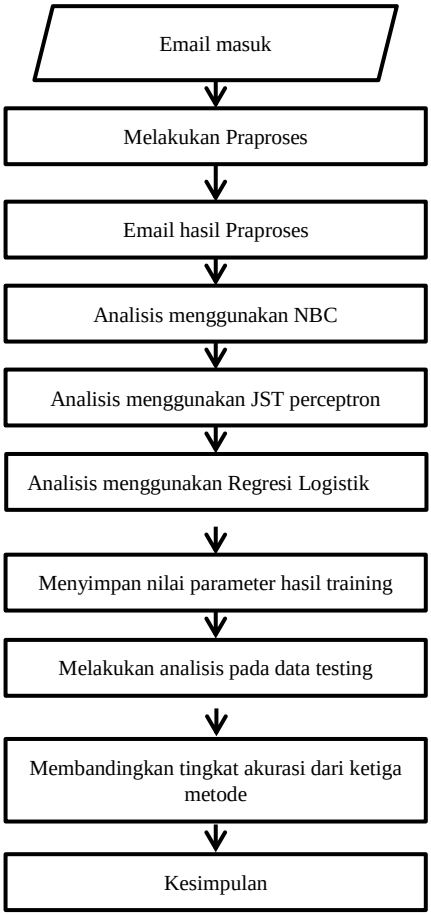
b) dengan targetnya ( jaringan sudah memahami pola). Iterasi tidak berhenti setelah semua pola dimasukan.
c) Perubahan bobot hanya dilakukan pada pola yang mengandung kesalahan (keluaran jaringan ≠ target). Perubahan tersebut merupakan hasil kali unit masukan dengan target laju pemahaman.
d) Nilai global optimum pada jaringan syaraf tiruan perceptron sangat sulit untuk dicapai, maka perlu dilakukan *running* secara berulang-ulang dengan mengubah nilai awal dari parameter-parameter dari *perceptron* itu sendiri.
e) Karena menggunakan metode pembelajaran *delta rule*, maka menggunakan nilai ambang batas 0.

4. Klasifikasi email menggunakan NBC dengan tahapan
   a) Membagi data menjadi *testing* dan *training*, pada data *training* telah diketahui jenis dari kategori email.
   b) Menghitung probabilitas prior dari $V_j$, dimana $V_j$ merupakan kagetori email, yaitu $j_1$ = Non-*Ads*, $j_2$ = *Ads.*
   c) Menghitung probabilitas kata $w_k$ pada kategori $v_j$.
   d) Model probabilitas NBC disimpan dan digunakan untuk tahap data *testing*.
   e) Menghitung probabilitas tertinggi dari semua kategori yang diujikan ($V_{MAP}$).
   f) Mencari nilai $V_{MAP}$ paling maksimum dan memasukkan email tersebut pada kategori dengan $V_{MAP}$ maksimum.
   g) Melakukan intervensi terhadap probabilitas prior agar didapatkan tingkat *FP Ratio* yang lebih kecil.
   h) Menghitung nilai akurasi dari model yang terbentuk.

`

5. Klasifikasi email menggunakan Regresi Logistik dengan tahapan
   a) Membagi data menjadi *testing* dan *training*, pada data training telah diketahui jenis dari kategori email.
   b) Melakukan uji independensi dengan menggunakan data training.
   c) Membuat model regresi logistik antara variabel bebas dan variabel terikat kemudian melakukan pengujian serentak dan parsial terhadap model yang diperoleh.
   d) Mengintepretasi model logistic biner dan juga odds ratio yang diperoleh.
   e) Menghitung ketepatan klasifikasi model regresi logistik.
6. Membandingkan performansi metode jaringan syaraf *Perceptron*, NBC, dan Regresi Logistik berdasarkan tingkat akurasi ketepatan klasifikasi, *False positive ratio* dan waktu yang digunkana untuk *running*.

   Langkah analisis ini apabila digambarkan dengan diagram alir maka akan nampak seperti berikut:




**Gambar 3.1** Diagram Alir Langkah Penelitian

`

*(Halaman sengaja dikosongkan)*

# BAB IV
# HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai klasifikasi dengan *Naive Bayes Classifier*, Jaringan Syaraf Tiruan *Perceptron* dan Regresi Logistik menggunakan data email. Sebelum menganalisis, dilakukan *pre processing text* pada email. Contoh data sebelum praproses dapat dilihat pada Lampiran 1.

## 4.1 Praproses Teks

Data email yang telah dikumpulkan kemudian dilakukan praproses yaitu menentukan *wordlist* dan *Feature Vector Content* menggunakan teks python script dengan bantuan *software* Sublime Text 2 (Unregistered Version). Sedangkan untuk *Feature Words Selection* menggunakan *software* R 3.1.3. Praproses pada keseluruhan text email yang menghasilkan *wordlist* disajikan pada Tabel 4.1.

Tabel 4.1 memperlihatkan 100 kata yang merupakan bagian dari *wordlist,* jumlah keseluruhan kata pada *wordlist* terlalu banyak sehingga tidak bisa terbaca seluruhnya oleh *software,* sehingga hanya ditampikan sampel dari kata yang merupakan bagian dari *wordlist*. *Wordlist* inilah yang kemudian digunakan sebagai variabel bebas. Seluruh kata dalam *wordlist* adalah kata yang tanpa melalui proses *stemming* (pembentukan kata dasar) sehingga kata yang terbentuk sebagai variabel bukan berbentuk kata dasar. Tidak juga dilakukan penghilangan kata penghubung, karena dalam persoalan yang menyangkut email, kata penghubung bisa dijadikan sebagai variabel yang berguna atau signifikan. Untuk Kemudian dari keseluruhan kata di dalam *wordlist* dihitung bobot *adicity*-nya. Dari bobot *adicity* kemudian disaring kembali untuk menemukan kata dengan bobot yang sesuai (lebih lengkapnya lihat pada Sub Bab 2.5). Pada Tabel 4.2,

kata telah dihitung bobotnya dan dipilih kata dengan bobot yang sesuai dengan *Feature Vector Content* yang diperlukan untuk dilakukan klasifikasi. *Feature Vector Content* dengan keseuruhan bobot bisa dilihat pada Lampiran 2.

**Tabel 4.1** *Wordlist*

| **Wordlist** | | | | |
|---|---|---|---|---|
| **Kata** | **Kata** | **Kata** | **Kata** | **Kata** |
| Xls | On | forwarded | 01 | Ramp |
| 11 | here | Hou | By | Ads |
| Com | please | No | Us | Illu |
| 02 | nom | http | questions | 2000 |
| Re | should | For | prices | enron |
| Volumes | 000 | Am | contract | ! |
| Link | our | Will | know | thanks |
| Cc | 2001 | One | online | 10 |
| Daren | farmer | The | 00 | most |
| Hpl | attached | ( | from | Stop |
| Your | me | Mmbtu | 03 | mary |
| Gas | meter | Email | www | Ect |
| Pm | corp | Be | volume | Sitara |
| Deal | more | If | need | 12 |
| Is | 09 | Flow | software | gex |
| Following | just | 06 | 30 | Let |
| Click | offers | Day | than | 31 |
| 04 | $ | Offer | bob | available |
| Free | net | 2004 | nomination | 07 |
| You | money | Microsoft | call | this |

Dari Tabel 4.1 kemudian dihitung bobotnya, berikut adalah daftar kata dengan bobot *Feature Vector Content* yang telah dihitung disajikan pada Tabel 4.2.

**Tabel 4.2** *Feature Vector Content* pada kata dalam *Wordlist*

| **Feature Vector Content** | | | | |
|---|---|---|---|---|
| **Kata** | **Non-Ads** | **Ads** | **Both** | **Adicity** |
| ramp | 0.711 | 0.000 | 0.711 | 0.000 |
| Ads | 0.003 | 0.487 | 0.484 | 0.994 |
| Illu | 0.000 | 0.453 | 0.453 | 1.000 |
| 2000 | 0.389 | 0.020 | 0.369 | 0.049 |
| enron | 0.360 | 0.000 | 0.360 | 0.000 |
| ! | 0.143 | 0.527 | 0.384 | 0.787 |
| thanks | 0.377 | 0.027 | 0.350 | 0.006 |
| Cc | 0.337 | 0.007 | 0.330 | 0.019 |
| daren | 0.311 | 0.000 | 0.311 | 0.000 |
| Hpl | 0.306 | 0.000 | 0.306 | 0.000 |
| your | 0.240 | 0.533 | 0.293 | 0.690 |
| Gas | 0.303 | 0.013 | 0.290 | 0.042 |
| Pm | 0.289 | 0.000 | 0.289 | 0.000 |
| deal | 0.271 | 0.007 | 0.265 | 0.024 |
| Ect | 0.263 | 0.000 | 0.263 | 0.000 |
| forwarded | 0.251 | 0.000 | 0.251 | 0.000 |
| Hou | 0.257 | 0.007 | 0.250 | 0.025 |
| No | 0.109 | 0.353 | 0.245 | 0.765 |
| http | 0.046 | 0.287 | 0.241 | 0.862 |
| : | : | : | : | : |
| : | : | : | : | : |

Dari seluruh kata dalam *Feature Vector Content* dipilih 100 kata dari urutan teratas yang kemudian akan dijadikan

`

variabel prediktor atau *word vector*, dari urutan teratas terpilih kata dengan bobot tertinggi pada Tabel 4.3.

**Tabel 4.3** Feature Word Selection

| **Word Vector (Word Selection)** | |
|---|---|
| **Kata** | **Kata** |
| ramp | ! |
| ads | Cc |
| illu | http |
| 2000 | Hpl |
| enron | your |
| thanks | pm |
| hpl | forward |
| gas | daren |
| deal | ect |
| hou | http |
| forwarded | day |
| no | offers |
| : | : |
| : | : |

Selanjutnnya kata dengan jumlah huruf kurang dari 2 akan ditambahkan suatu pengenal agar dapat terbaca oleh *software R* saat dilakukan pemanggilan *word vector*. Hasil keseluruhan bisa dilihat pada Lampiran 2. Kemudian hasil yang didapat akan tampak seperti berikut :

**Tabel 4.4** Penambahan Suatu Pengenal Pada Kata

| **Kata** | | **Hasil** |
|---|---|---|
| ! | --> | char_! |
| Cc | --> | som_cc |

Pada Tabel 4.4 kata "!" akan dikenali dengan nama char_! Dan kata "cc" akan dikenali dengan nama som_cc.

Dari Tabel 4.4 dicari frekwensi kemunclan kata tiap kategori email. Berikut merupakan frekwensi kata yang muncul pada tiap kategori disajikan pada Tabel 4.5.

**Tabel 4.5** Frekuensi Kemunculan Kata Pada Email

| **Non-Ads** | **Jumlah** | **Iklan (Ads)** | **Jumlah** |
|---|---|---|---|
| 2000 | 4290 | 2000 | 78 |
| Ads | 2 | Ads | 1900 |
| char_! | 1083 | char_! | 2459 |
| Enron | 6293 | Enron | 0 |
| Hpl | 2416 | Hpl | 0 |
| http | 233 | http | 983 |
| Illu | 0 | Illu | 1201 |
| Ramp | 27311 | Ramp | 2 |
| som_cc | 1719 | som_cc | 12 |
| Your | 1867 | Your | 1952 |
| : | : | : | : |
| : | : | : | : |

Dalam uji *Naive Bayes Classifier* (NBC), Jaringan Syaraf Tiruan *Perceptron* maupun Regresi Logistik ini akan dibagi menjadi dua partisi data yaitu data *training* dan data *testing*. Tujuan dari pembagian data menjadi beberapa partisi data adalah untuk mengtahui apakah ada perbedaan pada hasilnya, dan untuk membandingkan pada partisi manakah akurasi terbaik diperoleh. Frekwensi pada tabel 4.5 tidak ditampilkan seluruhnya, hasil keseluruhan pada Tabel 4.5 selengkapnya pada Lampiran 2. Pada data *training* dan *testing* masing-masing pembagiannya adalah 50:50, 60:40, 70:30, 80:20, 90:10, dan mengikuti pembagian partisi yang disajikan pada Tabel 4.6.

`

**Tabel 4.6** Partisi data *Training* dan *Testing*

| Partisi |
|---|
| 50 : 50 |
| 60 : 40 |
| 70 : 30 |
| 80 : 20 |
| 90 : 10 |

Dalam percobaan ini untuk mendapatkan hasil klasifikasi terbaik akan digulirkan tiap-tiap partisi untuk pengambilan data *training* maupun *testing*. Untuk data hasil Praproses bisa dilihat pada Lampiran 3.

## 4.2 *Naive Bayes Classifier*

*Naive Bayes* merupakan suatu metode klasifikasi dengan menggunakan probabilitas keanggotaan dalam suatu kelas. Pada penelitian ini, *Naive Bayes* digunakan untuk mengklasifikasikan sebuah email berisi iklan (Ads) atau tidak. Langkah awal dalam klasifikasi menggunakan Naive Bayes ialah menghitung nilai *prior probability* untuk setiap kategori pada variabel respon. Berikut nilai *prior probability* disajikan pada Tabel 4.6 :

**Tabel 4.7** Prior Probability untuk Tiap Variabel Respon

| Kategori | Non-Ads | Ads |
|---|---|---|
| *Prior Probability* | 0.695 | 0.305 |

Berdasarkan Tabel 4.7 diketahui *prior probability* bukan iklan lebih besar dibandingkan *prior probability* email iklan. Selanjutnya, prosedur pengklasifikasian dilakukan dengan memaksimumkan nilai *posterior probability*, dimana terlebih dahulu dihitung nilai *conditional probability* berdasarkan nilai

*mean* dan *variance*. Jika nilai $P(Y = Non - Ads|\boldsymbol{X})$ lebih besar dibandingkan nilai $P(Y = Ads|\boldsymbol{X})$ maka pengamatan dikelompokkan dalam kelompok email bukan iklan. Sedangkan jika nilai $P(Y = Ads|\boldsymbol{X})$ lebih besar dibandingkan nilai $P(Y = Non - Ads|\boldsymbol{X})$ maka pengamatan dikelompokkan dalam kelompok email iklan.

### 4.2.1 Pengukuran Performa NBC

Seperti yang telah disebutkan pada praproses data bahwa data akan dipartisi menjadi 5 *part* atau bagian, maka setelah itu akan dilakukan analisis pada tiap bagian tersebut untuk mendapatkan hasil terbaik dari klasifikasi menggunakan NBC, yang pertama akan dibahas adalah NBC pada partisi data 50 : 50 untuk *training* dan *testing* seperti pada table berikut :

**Tabel 4.8** Ketepatan Klasifikasi NBC pada partisi data 50 : 50

| | **Training** | | **Testing** | |
|---|---|---|---|---|
| | **Ads** | **Non-Ads** | **Ads** | **Non-Ads** |
| **Ads** | 711 | 126 | 809 | 126 |
| **Non-Ads** | 18 | 1673 | 11 | 1582 |
| **Error** | | 0.056962 | | 0.054193 |

Berdasarkan Tabel 4.8 nilai error pada data training 0.056962 atau 5,6% dan 0.054193 atau 5,4% pada data testing. hasil klasifikasi tersebut sudah dirasa cukup baik, namun yang menjadi permasalahan adalah masih tingginya *false positive* (email bukan iklan yang terklasifikasi sebagai email iklan). Dengan jumlah *false positive* pada data training sebesar 126 email dan pada data testing juga 126 email, yang apabila dihitung nilai *false positive ratio* -nya sebesar 0.070 atau 7% untuk data *training* dan 0.074 atau 7,4% pada data *testing.* Partisi-partisi data

`

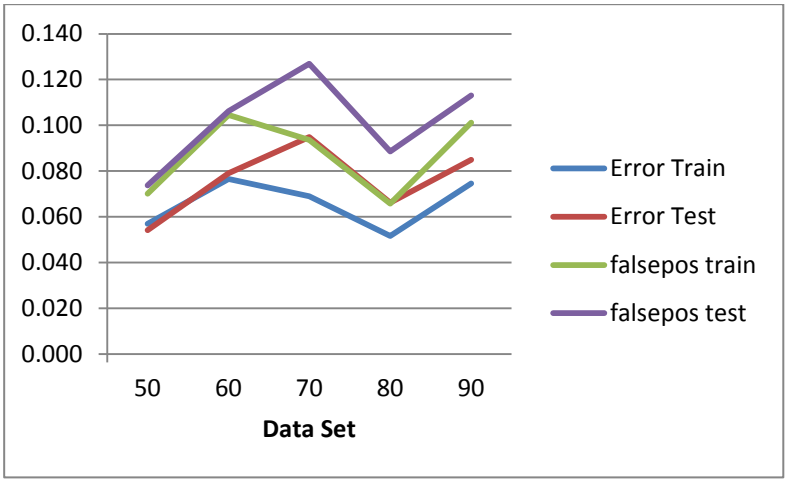
selanjutnya yaitu 60 : 40, 70:30, 80:20, 90:10 akan dihitung dengan cara yang sama dan menghasilkan error dan ketepatan klasifikasi, yang kemudian hasil dari keseluruhan partisi dijabarkan pada Tabel 4.9 di bawah ini :

**Tabel 4.9** Performa NBC pada Tiap Partisi

| | **Training** | | **Testing** | |
|---|---|---|---|---|
| **Dataset** | **Error Train** | **Akurasi** | **Error Test** | **Akurasi** |
| **50 : 50** | 0.057 | 0.943 | *0.054 | 0.946 |
| **60 : 40** | 0.076 | 0.924 | 0.079 | 0.921 |
| **70 : 30** | 0.069 | 0.931 | 0.095 | 0.905 |
| **80 : 20** | *0.052 | 0.948 | 0.066 | 0.934 |
| **90 : 10** | 0.075 | 0.925 | 0.085 | 0.915 |

| **Dataset** | **falsepos train** | **falsepos test** |
|---|---|---|
| **50 : 50** | 0.070 | *0.074 |
| **60 : 40** | 0.105 | 0.106 |
| **70 : 30** | 0.094 | 0.127 |
| **80 : 20** | *0.066 | 0.089 |
| **90 : 10** | 0.101 | 0.113 |

Dari Tabel 4.9 di atas, dapat diketahui bila akurasi terbaik pada data *training* diperoleh pada partisi 80:20, dan 50:50 pada data *testing,* dan pada partisi yang sama untuk *false positive ratio.* Sedangkan nilai akurasi terendah untuk data *training* terletak pada partisi 60:40 dan 70:30 untuk data *testing* dan partisi yang sama untuk *false positive ratio.*. Untuk hasil lengkap bisa dilihat pada Lampiran 4. Untuk melihat lebih jelas bagaimana performa NBC dalam mengklasifikasikan tiap partisi data, dapat dilihat pada Gambar 4.1 berikut :



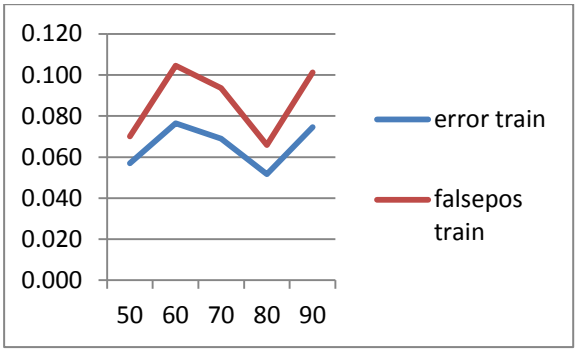

**Gambar 4.1** Peforma NBC pada Tiap Partisi

Pada Gambar di atas memperlihatkan bahwa pola tingkat error naik pada partisi 60:40 dam 70:30 dan turun pada partisi 80:20. Pada Gambar 4.1 juga menunjukkan bahwa *false positive ratio* lebih tinggi ketimbang error *rate* pada data *training* maupun *testing*, ini menunjukkan bahwa klasifikasi NBC belum sepenuhnya baik karena masih mengandung *false positive ratio* yang cukup tinggi.

### 4.2.2 Intervensi Probabilitas Prior

Dapat diketahui pada Gambar 4.1 bahwa nilai *false positive ratio* pada data *training* dan *testing* masih tinggi dan melebihi error *rate*, hal ini harus ditanggulangi mengingat *false positive* (email bukan iklan yang terklasifikasi sebagai email iklan) pada penelitian berkaitan dengan email selalu diupayakan untuk mencapai angka 0 dengan rasio 0%. Untuk lebih jelas dalam menggambarkan bagaimana tingginya jumlah *false positive* dan *false positive ratio* pada data training,bisa dilihat pada Gambar 4.2.

`




**Gambar 4.2** Error dan False Positive Ratio pada Data Training

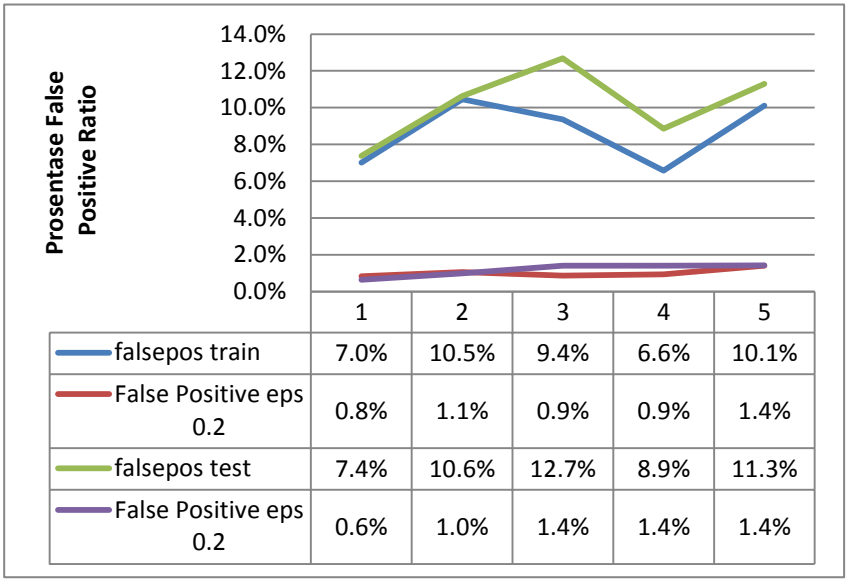
Pada Gambar 4.2 dapat diketahui bahwa *false positive ratio* lebih tinggi dari nilai error sendiri, *false positive ratio* yang paling rendah diperoleh dari partisi 80:20 yaitu sebesar 6,6% yang berarti bila ada email yang bukan iklan masuk sebesar 1000 email, maka akan tertolak sebesar 66 email. Jika email yang terlolak merupakan email penting yang harusnya terbaca oleh pengguna, maka akan timbul masalah, maka diperlukan penanggulangan terhadap tingginya *false positive ratio*.

Untuk menanggulangi permasalahan pada tingginya jumlah *false positive* terdapat beberapa langkah yang bisa dilakukan antara lain, merubah *threshold*, *epsilon range*, dan melakukan *laplacian smoothing* yang ketiganya akan berdampak pada berubahnya probabilitas prior, dengan berubahnya probailitas prior maka probabilitas posterior juga akan mengalami perubahan yang nantinya akan berdampak pada proses klasifikasi. Dalam penelitian kali ini hanya akan digunakan perubahan nilai *epsilon range* yaitu sebesar 0.2, nilai 0.2 digunakan karena dari *trial* dan *error* yang telah dilakukan nilai 0.2 pada *epsilon range* memberikan efek yang paling signifikan untuk meminimalisir jumlah *false positive* dan *false positive ratio*. Berikut adalah hasil dari klasifikasi dengan NBC pada data *training* dan *testing* untuk pastisi data 50 : 50 disajikan pada tabel 4.9.

**Tabel 4.10 NBC partisi 50 : 50 dengan Epsilon Range 0.2**

| | Training | | Testing | |
|---|---|---|---|---|
| | Ads | Non-Ads | Ads | Non-Ads |
| **Ads** | 504 | 15 | 570 | 11 |
| **Non-Ads** | 225 | 1784 | 250 | 1697 |
| **Error** | | 0.094937 | | 0.103244 |
| **false pos** | | 0.008338 | | 0.00644 |

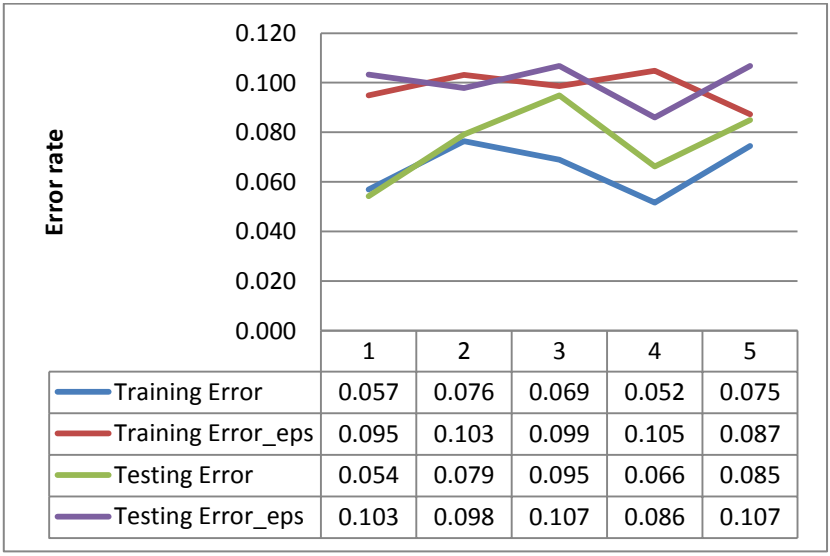
Dari Tabel 4.10 di atas, nilai angka *false positive* bisa ditekan dengan sangat signifikan menjadi hanya 15 email pada data *training* dan 11 email pada data *testing*. Untuk lebih jelasnya dalam melihat bagaimana peran intervensi probabilitas prior pada data *training* dan *testing* untuk menekan jumlah *false positive* pada tiap-tiap partisi, bisa dilihat pada Gambar 4.3 di bawah ini :




**Gambar 4.3** False Positive Ratio dengan Epsilon Range 0.2

`

Dapat dilihat pada Gambar 4.2, jumlah *false positive* bisa ditekan dari 126 menjadi hanya 15 email pada data training di partisi 50 : 50. Namun jumlah *false positive* yang turun dengan signifikan menghasilkan nilai error yang semakin bertambah, bisa dilihat pada Gambar 4.4 di bawah ini :




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Training Error | 0.057 | 0.076 | 0.069 | 0.052 | 0.075 |
| Training Error_eps | 0.095 | 0.103 | 0.099 | 0.105 | 0.087 |
| Testing Error | 0.054 | 0.079 | 0.095 | 0.066 | 0.085 |
| Testing Error_eps | 0.103 | 0.098 | 0.107 | 0.086 | 0.107 |

**Gambar 4.4** Error Rate dengan Epsilon Range 0.2

Dari Gambar 4.4 di atas diperoleh hasil pada partisi 50 : 50 pada data *training* nilai error yang semula 5,7% naik menjadi 9.5%, dan seterusnya. Hal ini menunjukkan bahwa penurunan jumlah *false positive* dibarengi dengan naiknya error, atau turunnya *false positive ratio* dibarengi dengan naiknya prosentasie error. Secara lengkap performa NBC dengan *epsilon range* sebesar 0.2 adalah sebagai berikut :

**Tabel 4.11** Performa NBC dengan Epsilon Range 0.2

| | Training | | Testing | |
|---|---|---|---|---|
| **Dataset** | **Error** | **Akurasi** | **Error** | **Akurasi** |
| **50 : 50** | 0.095 | 0.905 | 0.103 | 0.897 |
| **60 : 40** | 0.103 | 0.897 | 0.098 | 0.902 |
| **70 : 30** | 0.099 | 0.901 | 0.107 | 0.893 |
| **80 : 20** | 0.105 | 0.895 | 0.086 | *0.914 |
| **90 : 10** | *0.087 | 0.913 | 0.107 | 0.893 |
| **Dataset** | **False Positive Ratio** | | **False Positive Ratio** | |
| **50 : 50** | 0.0083 | | 0.0064 | |
| **60 : 40** | 0.0105 | | *0.0099 | |
| **70 : 30** | *0.0008 | | 0.0140 | |
| **80 : 20** | 0.0093 | | 0.0141 | |
| **90 : 10** | 0.0140 | | 0.0141 | |

Dari Tabel 4.11 di atas nilai error terendah untuk NBC dengan epsilon range sebesar 0.2 terletak pada partisi 90 : 10 pada data *training* dan partisi 80 : 20 pada data *testing*. Sedangkan untuk *false positive ratio* terletak pada partisi 70 : 30 pada data *training* dan 60 : 40 pada data *testing*. *False Positive Ratio* terendah mencapai angka 0.08%, sudah sangat kecil hingga mendekati nol. Untuk data lengkap performa Naïve Bayes dengan Epsilon Range 0.2 bisa dilihat pada Lampiran 4.

### 4.2.3 Model *Naïve Bayes Classifier*

Berdasarkan persamaan (2.10) berikut merupakan bentuk persamaan dari NBC untuk tiap kategori email dengan menggunakan *word vector* sebanyak 10 yang diperoleh dari pembobotan (lihat Tabel 4.3). Artinya jumlah $a_i$ adalah sebanyak 10 dan $v_j$ sebanyak kategori yaitu 2. Misalkan pada partisi 50:50, data training yang digunakan 50% atau setengah dari seluruh data, maka nilai $P(v_j)$ sebesar 0.695 untuk kategori email bukan

`

iklan dan 0.305 untuk kategori email iklan. Persamaan di bawah menunjukkan persamaan berurutan menggunakan kata “2000”, “ads”, “char_!”, dan seterusnya pada kategori email bukan iklan dan email iklan.

$$V_{MAP} = \underset{v_j \in V}{\operatorname{argmax}}\ P\left(v_j\right)\prod\nolimits_i P\left(a_i \mid v_j\right)$$

Persamaan Kategori Email Non-iklan :

$$(0,695)\times(9.48E-02)\times(6.63E-05)\times(2.39E-02)\times\cdots\times(4.13E-02)$$

Persamaan Kategori iklan :

$$(0,305)\times(9.18E-03)\times(2.21E-01)\times(2.86E-01)\times\cdots\times(2.27E-01)$$

Apabila ada sebuah email masuk maka probabilitas tertinggi diantara 2 kategori tersebut yang akan dipilih. Perhitungan probabilitasnya adalah apabila sebuah email memiliki kemunculan kata sebanyak satu kali maka kata tersebut pada persamaan akan dipangkatkan 1. Sedangkan kata yang tidak ada atau tidak muncul maka pada persamaan probabilitas kata tersebut akan dipangkatkan 0. Untuk probabilitas kata ke-i pada tiap kategori dapat dilihat pada lampiran.

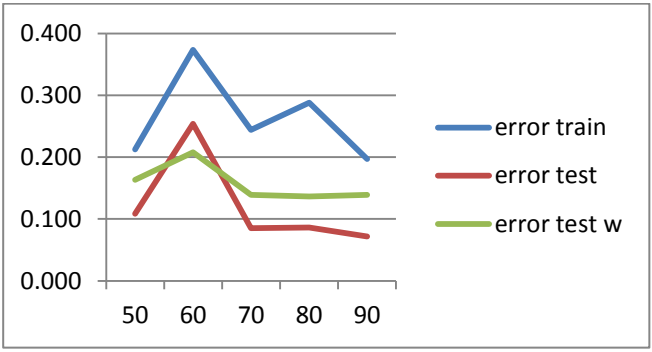
## 4.3 *Perceptron*

Algororitma pada perceptron merubah dan meng-*update* bobot hingga bobot tidak mengalami perubahan lagi, sehingga dari partisi data (Tabel 4.6) untuk tiap partisi akan menghasilkan bobot untuk tiap *word vector* yang digunakan, yang diharapkan mampu merepresentasikan sebuah fungsi pemisah untuk kedua kategori email. Setelah bobot dari hasil *training* diperoleh, bobot dapat langsung digunakan untuk memisahkan data pada data *testing,* tanpa meng-*update* lagi bobotnya pada tiap epoch *testing,* namun akan menarik bila pada data *testing* bobot hasil *training* kembali di-*update*. Hasil klasifikasi *training testing* untuk tiap partisi data dengan menggunakan perceptron dengan bobot awal 0 bisa dilihat pada Tabel 4.12 berikut ini :

**Tabel 4.12** Performa Perceptron dengan Bobot Awal 0

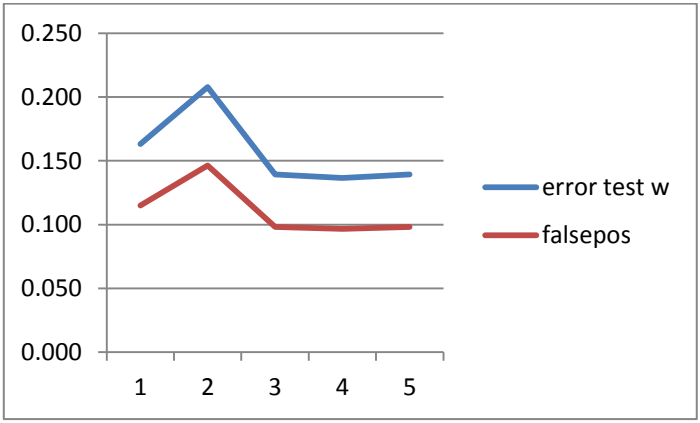
| Dataset | error train | Akurasi | error test | Akurasi |
|---|---|---|---|---|
| **50 : 50** | 0.213 | 0.787 | 0.108 | 0.892 |
| **60 : 40** | 0.374 | 0.627 | 0.254 | 0.746 |
| **70 : 30** | 0.244 | 0.756 | 0.085 | 0.915 |
| **80 : 20** | 0.288 | 0.712 | 0.086 | 0.914 |
| **90 : 10** | *0.197 | 0.803 | *0.072 | 0.928 |
| **Dataset** | **error test w** | **Akurasi** | **falsepos** | |
| **50 : 50** | 0.163 | 0.837 | 0.115 | |
| **60 : 40** | 0.208 | 0.792 | 0.146 | |
| **70 : 30** | 0.139 | 0.861 | 0.098 | |
| **80 : 20** | *0.136 | 0.864 | *0.097 | |
| **90 : 10** | 0.139 | 0.861 | 0.098 | |

Dari Tabel 4.12 di atas dapat diketahui bahwa nilai akurasi pada error tanpa meng-*update* bobot pada data *testing* lebih tinggi ketimbang melakukan *update* bobot pada tiap iterasi pelatihan dalam data *testing*. Dengan error test terendah pada partisi 90 : 10 yaitu 0.072, sedangkan *false positive* terendah pada partisi data 80 : 20. Untuk menggambarkan bagaimana *update* bobot mempengaruhi klasifikasi dilihat pada Gambar 4.5.




**Gambar 1.5** Error Test Tanpa dan Dengan *Update* Bobot

Gambar 4.5 memperlihatkan bahwa error *testing* tanpa melanjutkan *update* bobot (menggunakan bobot hasil *training*) lebih kecil dibandingkan melanjutkan *update* bobot. Error pada *testing* tanpa meng-*update* bobot cenderung mengikuti pola dari error pada *training*. Untuk melihat bagaimana hasil *false positive* pada *testing* disajikan pada Gambar 4.6.




**Gambar 4.6** Error Test dan False Positive Ratio

Dengan melihat Gambar 4.6 bahwa false positive ratio telah mempunyai nilai yang cukup rendah dan berada di bawah error test.

### 4.3.1 Global Optimization pada Perceptron

Optimasi merupakan usaha yang dilakukan untuk memperoleh hasil akhir yang lebih baik, permasalahan optimasi terletak pada tujuan untuk mendapatkan atau menemukan solusi terbaik dari semua solusi yang mungkin. Untuk menemukan kemungkinan lain dalam memperoleh hasil klasifikasi, dalam penelitian ini akan digunakan nilai initial bobot atau bobot awal yang berbeda-beda, guna menemukan bobot awal mana yang memiliki pengaruh besar dalam meminimalisir error klasifikasi

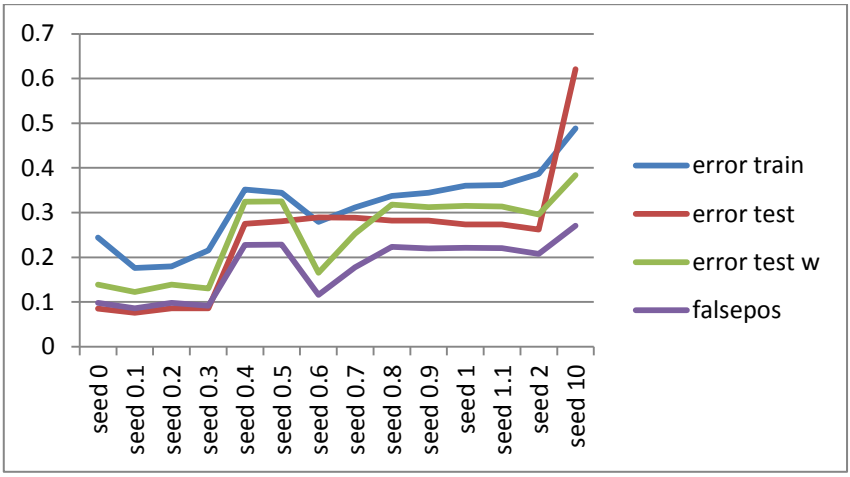
menggunakan jaringan perceptron. Untuk melihat bagaimana hasil dari beragam initial bobot atau bobot awal yang telah digunakan, bisa dilihat pada Tabel 4.13 berikut ini.

**Tabel 4.13** Performa Perceptron dengan Bobot Awal yang Berbeda-Beda (Partisi Data 70 : 30)

| | **seed ke 1** | **seed ke 2** | **seed ke 3** | **seed ke 4** | **seed ke 5** |
|---|---|---|---|---|---|
| **error train** | 0.2442 | *0.1762 | 0.1801 | 0.2155 | 0.3517 |
| **error test** | 0.0851 | *0.0760 | 0.0863 | 0.0863 | 0.2751 |
| **error test w** | 0.1392 | *0.1224 | 0.1392 | 0.1302 | 0.3241 |
| **falsepos** | 0.0980 | *0.0862 | 0.0980 | 0.0917 | 0.2278 |
| | **seed ke 6** | **seed ke 7** | **seed ke 8** | **seed ke 9** | **seed ke 10** |
| **error train** | 0.3445 | 0.2793 | 0.3119 | 0.3373 | 0.3445 |
| **error test** | 0.2809 | 0.2893 | 0.2887 | 0.2822 | 0.2822 |
| **error test w** | 0.3254 | 0.1656 | 0.2532 | 0.3177 | 0.3125 |
| **falsepos** | 0.2287 | 0.1162 | 0.1779 | 0.2232 | 0.2196 |
| | **seed ke 11** | **seed ke 12** | **seed ke 13** | **seed ke 14** | |
| **error train** | 0.3599 | 0.3616 | 0.3870 | 0.4887 | |
| **error test** | 0.2738 | 0.2738 | 0.2622 | 0.6211 | |
| **error test w** | 0.3151 | 0.3138 | 0.2957 | 0.3840 | |
| **falsepos** | 0.2214 | 0.2205 | 0.2078 | 0.2704 | |

Dari Tabel 4.13 diketahui bahwa error terkecil terletak pada bobot awal saat proses seed ke 2, dengan semua pengukuran error baik pada *training, testing,* serta *false positive* memiliki nilai error yang lebih kecil disbanding dengan perceptron dengan nilai bobot awal yang lain. Untuk hasil lengkap dari Perceptron bisa dilihat pada Lampiran 5. Untuk lebih jelas bisa dilihat pada Gambar 4.7.

`




**Gambar 4.7** Perceptron dengan Bobot Awal yang Berbeda-Beda (Partisi Data 70 : 30)

Pada penelitian ini optimasi hanya dilakukan pada penentuan awal bobot, dan perceptron dengan bobot awal dengan nilai error terendah adalah perceptron dengan bobot awal 0.1, maka yang akan digunakan sebagai bobot akhir untuk menentukan model adalah bobot akhir yang dihasilkan perceptron dengan bobot awal sebesar 0.1 dengan partisi data sebesar 70 : 30. Berikut ditampilkan 100 bobot akhir perceptron dengan menggunakan bobot awal dari *seed* atau hasil run ke 1 dan partisi data 70 : 30 pada tabel 4.14.

Tabel 4.14 berisi koefisian bobot *synaptic perceptron* yang nantinya digunakan dalam pemodelan *perceptron.* Model *perceptron* yang didapatkan adalah model yang telah dilakukan usaha untuk mengoptimalkan (dengan mengubah bobot awal) secara global.

**Tabel 4.14** Bobot Akhir yang Didapat untuk Perceptron dengan Bobot Awal 0.1 dan Partisi Data 70 : 30.

| **Bobot** | | **Bobot** | |
|---|---|---|---|
| **w1** | | -0.06 | |
| **w2** | -1.2862 | **w7** | -0.729 |
| **w3** | -0.9686 | **w8** | 0.456 |
| **w4** | 0.8595 | **w9** | -0.7805 |
| **w5** | -0.2282 | **w10** | -0.9213 |
| **w6** | -0.8263 | **w11** | 1.0004 |
| **:** | : | **:** | : |
| **:** | : | **:** | : |

Dari Tabel 4.14 didapatkan 100 bobot akhir untuk perceptron dengan bobot awal 0.1. Dengan w1 merupakan *dummy* variabel yang merupakan bias. Bobot-bobot inilah yang nantinya akan digunakan dalam menentukan model jaringan perceptron.

### 4.3.2 Model Jaringan Syaraf Tiruan Perceptron

Model pada output jaringan syaraf tiruan perceptron untuk dua kategori email dengan menggunakan metode pembelajaran *delta rule*, metode pembelajaran delta rule yang tidak mengandung nilai batas pada fungsi aktivasinya menyebabkan output target menjadi -1 dan 1 (bipolar) dengan batas 0. Maka persamaan yang bisa dituliskan untuk kedua kategori adalah sebagai berikut.

$$net = -0.06 - 1.2862x_1 - 0.9686x_2 + 0.8595 \ldots - 0.4891x_{100}$$

Apabila ada email masuk kemudian dihitung nilai net, jika nilai net lebih besar dari 0 maka email masuk ke dalam kategori iklan atau *Ads* dan jika hasil net kurang dari atau sama dengan nol maka email masuk dalam kategori bukan iklan.

`

## 4.4 Regresi Logistik

Metode ketiga yang digunakan untuk klasifikasi email iklan adalah regesi logistik. Pada bab regresi logistik ini akan dihitung ketepatan klasifikasi seperti pada kedua metode sebelumnya.

### 4.4.1 Pengujian Signifikansi Parameter

Uji serentak adalah uji yang mempunyai fungsi dimana untuk mengetahui signifikansi parameter pada konstanta secara keseluruhan, berikut hasil uji serentak.

**Tabel 4.15** Uji Serentak Omnibus

| | Chi-square | df | Sig. |
|---|---|---|---|
| Step 1 | 491.388 | 100 | 0 |

Berdasarkan Tabel 4.15 diperoleh nilai signifikansi model sebesar 0.000 karena nilai ini lebih kecil dari 5% maka disimpulkan bahwa variabel bebas yang digunakan, secara bersama-sama berpengaruh terhadap jenis *email* terhadap variabel prediktornya atau ada salah satu variabel prediktor yang berpengaruh.

### 4.4.2 Koefisien Parameter dan Model Regresi Logistik

Setelah melakukan pengujian serentak dan diperoleh hasil bahwa variabel prediktor berpengaruh, maka dilanjutkan untuk mengetahui nilai dari koefisien dari tiap-tiap variabel yang nantinya akan digunakan untuk pembentukan model, dengan hasil seperti Tabel 4.16 di bawah ini.

**Tabel 4.16** Koefisien Parameter Regresi Logistik

| | Coef | Exp(Coef) |
|---|---|---|
| Constant | -3.893 | 0 |
| 2000 | 0.232 | 1.285 |
| Ads | -0.133 | 0.339 |
| char_! | 0.436 | 1.399 |
| enron | 0.519 | 1.445 |
| hpl | 0.038 | 1.214 |
| http | 0.188 | 1.474 |
| illu | -0.102 | 0.852 |
| ramp | 0.345 | 1.556 |
| som_cc | 0.504 | 1.156 |
| : | : | : |
| : | : | : |

Exp (Coef) pada observasi pertama Tabel 4.16 artinya bahwa email yang memiliki kata 2000 cenderung masuk ke dalam kategori email bukan iklan sebesar 1.2 kali dibandingkan masuk ke dalam kategori email iklan. Begitu seterusnya. Hasil dari koefisien regresi logistic bisa dilihat pada Lampiran 6.

Model Regresi logistic yang terbentuk berdasarkan Tabel 4.15 adalah sebagai berikut.

$$g(x) = -3.893 + 0.232\,x1 - 0.133x2 + 0.436x3 + \ldots$$

dan model regresi logistiknya adalah :

$$\pi(x) = \frac{e^{-3.893 + 0.232\,x1 - 0.133x2 + 0.436x3 + \ldots}}{1 + e^{-3.893 + 0.232\,x1 - 0.133x2 + 0.436x3 + \ldots}}$$

Model dari regresi logistik di atas bisa disimpulkan bahwa peluang email untuk masuk kategori bukan iklan dipengaruhi oleh kata 2000 sebesar 0.232, dan seterusnya

`

### 4.4.3 Ketepatan Klasifikasi Regresi Logistik

Setelah dilakukan intepretasi koefisien dan telah diapatkan model regresi logistic, maka selanjunya dihitung ketepatan klasifikasi untuk tiap partisi data dan didapatkan hasil seperti berikut:

**Tabel 4.17** Ketepatan Klasifikasi Regresi Logistik

| Dataset | error train | Akurasi | error test | akurasi | falsepos |
|---|---|---|---|---|---|
| **50:50** | 0.190 | 0.810 | 0.098 | 0.902 | 0.151 |
| **60:40** | 0.120 | 0.880 | 0.212 | 0.788 | 0.198 |
| **70:30** | 0.180 | 0.820 | 0.129 | 0.871 | 0.102 |
| **80:20** | 0.108 | 0.892 | 0.106 | 0.894 | 0.197 |
| **90:10** | 0.099 | 0.901 | 0.110 | 0.890 | 0.102 |

Dari Tabel 4.17 dapat diketahui bahwa akurasi terbaik dari regresi logistik terdapat pada partisi 70:30 yaitu saat false positive ratio menyentuh angka terkecil.

## 4.5 Perbandingan Antara NBC, Perceptron, dan Regresi Logistik

Setelah mengetahui hasil masing-masing ketepatan klasifikasi pada ketiga metode maka langkah selanjutnya adalah membandingkannya. Berikut merupakan perbandingan antara kedua metode berdasarkan akurasi, dan *false positive ratio.* Untuk nilai yang diambil sebagi pembanding adalah nilai partisi yang memiliki *false positive ratio* terkecil.

Perbandingan Akurasi dan *False Positive Ratio* antara Naïve Bayes Classifier, Jaringan Syaraf Tiruan Perceptron, dan Regresi Logistik bisa dilihat pada tabel 4.18.

**Tabel 4.18** Perbandingan Hasil Ketepatan Klasifikasi Antara NBC dan Perceptron data *Testing*

| | **Testing** | | | |
|---|---|---|---|---|
| | **Tanpa Optimasi** | | **Optimasi NBC (Intervensi Prob Prior) Perceptron (Bobot Awal)** | |
| **Metode** | Akurasi | FP Ratio | Akurasi | FP Ratio |
| **NBC** | *0.946 | *0.074 | 0.902 | *0.009 |
| **Perceptron** | 0.928 | 0.098 | *0.924 | 0.0862 |
| **Reglog** | 0.871 | 0.102 | - | - |

Melihat hasil dari Tabel 4.18 maka NBC lebih unggul atau lebih baik dibanding Perceptron dan Regresi Logistik dalam mengklasifikasikan email iklan pada data *Testing* tanpa optimasi, namun akurasi NBC berada di bawah Perceptron pada data *Testing* dengan optimasi, hal ini dikarenakan *False Positive Ratio* NBC yang mampu ditekan sampai mendekati 0, yaitu 0,9%. Berdasarkan hasil di atas, pada NBC *False Positive Ratio* lebih mudah untuk dikontrol.

`

*(Halaman sengaja dikosongkan)*

# BAB V
# KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Setelah sebelumnya didapatkan hasil dari ketiga metode, berikut merupakan kesimpulan yang didapatkan.

1. Metode *Naive Bayes Classifier* dapat melakukan klasifikasi email iklan dan non iklan dengan sangat baik. Hasil akurasi tertinggi yang didapatkan pada saat data *testing* tanpa intervensi *probabilitas prior* adalah 94,6% dengan *False Positive Ratio* 7,4%. Dan dengan intervensi Probabilitas Prior menghasilkan akurasi 90,2% dan *False Positive Ratio* 0,9%.
2. Perceptron dalam melakukan klasifikasi email juga menghasilkan akurasi yang cukup baik. Menggunakan data *testing* tanpa optimasi didapatkan Akurasi 92,8% dan *False Positive Ratio* 9,8% dan dengan optimasi menghasilkan akurasi 92,4% dan *False Positive Ratio* 8,62%.
3. Regresi Logistik memiliki tingkat *false positive ratio* tertinggi pada partisi data 70 : 30, yaitu sebesar 0.102.
4. Hasil penelitian menunjukkan bahwa NBC lebih unggul dibanding Perceptron, dan regresi logistic. Pada NBC False Positive Ratio lebih mudah untuk dikontrol.

## 5.2 Saran

Saran untuk penelitian yang akan datang adalah.

1. Dalam penelitian klasifikasi email ini nilai epsilon range pada NBC diperoleh dari hasil *trial and error,* untuk penelitian selanjutnya dirasa perlu untuk menghitung nilai epsilon range yang sesuai pada data.
2. Membentuk *GUI* (Graphical User Interface) agar menjadi suatu bentuk otomatisasi yang benar-benar bisa diterapkan dalam kehidupan sehari-hari.

*(Halaman sengaja dikosongkan)*

## DAFTAR PUSTAKA


Ahmad, A. M., Ismail, S., & Samaon, D. F. (2004). Recurrent Neural Network with Backpropagation through Time for Speech Recognition. *International Symposium on Communications and Information Technologies*. Oktober 26-29

Agresti, A., (2002). *Categorical Data Analysis Second Edition*. New York: John Wiley & Sons.

Alo, Liliweri. (2011). *Komunikasi Serba Ada Serba Makna. Jakarta. Kencana Prenada Media Group.*

Anugroho, Prastyo & Winarno Idris (2012). Klasifikasi Email Spam dengan Menggunakan Metode Naïve Bayes Classifier Menggunakan Java Programming. *Politeknik Negeri Surabaya.*

Asian, J. A. (2007). Stemming Indonesian : A Confix-Stripping Approach. *ACM Trnsactions on Asian Language Information Processing (TALIP), 6(4),* , 1-33.

Buana, P. W., Jannet, S., & Putra, I. G. (2012). Combination of K-Nearest Neighbor and K-Means based on Term Re-weighting for Classify Indonesian News. *Journal of Communication Technology.*

Darujati, C., & Gumelar, A.B. (2012). Pemanfaatan Teknik Supervised Untuk Klasifikasi Teks Bahasa Indonesia, *Jurnal Bandung Text Mining*, 16(1),pp.5-1 – 5-8.

Dragut, Fang, Sistla, Yu, & Meng, (2009). *Stop Word and Related Problems in Web Interface Integration.*

Fausett, Laurene. (1994). *Fundamentals of Neural Networks Architectures, Algorithms, and Applications*. London: Prentice Hall, Inc.

Hermawan, A. (2006). *Jaringan Syaraf Tiruan: Teori dan Aplikasi.* Yogyakarta: ANDI

Hosmer, David W. & Lemeshow, Stanley. (2000). *Applied Logistik Regression*. Willey

IronPort System, Inc. (2006) "Spammers Continue Innovation:IronPort Study Shows Images Based Spam, Hit & Run, and Increased Volume Latest Thread to Your Inbox". *Journal of IronPort System.*

Kasali, Rhenald. (1995). *"Manajemen Periklanan"*. Pustaka Grafiti, Jakarta.

Kufadinimbwa, Owen & Gotora, Richard (2012). Spam Detection Using Artificial Neural Networks (Perceptron Leraning Rule). *Department of Computer Science, Faculty of Sciences, University of Zimbabwe.*

Kurniawan, B., Effendi, S., & Sitompul, O. S. (2012). Klasifikasi Konten Berita Dengan Metode. *JURNAL DUNIA TEKNOLOGI INFORMASI Vol. 1, No. 1*, 14-19.

Kusumadewi, S. (2003). *Artificial Intelligence (Teknik dan Aplikasinya), Membangun Jaringan Syaraf Tiruan (Menggunakan Matlab dan Excel Link)*. Yogyakarta: Graha Ilmu

Lestari, N. A., Putra, I. G., & Cahyawan, A. A. (2013). Personality Types Classification for Indonesian Text in Partners Searching Website Using Naïve Bayes Methods. *IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 3*, 1-8.

Liliana, D. Y., Hardianto, A., & Ridok, M. (2011). Indonesian News Classification using Support Vector Machine. *World Academy of Science, Engineering and Technology Vol:5* , 621-624.

Messaging Anti-Abuse working Group, (2006), "Email Metrics Program: The Network Operator's Perspective". *4th Quarters 2005 report.*

Miller, T. (2005). *Data and Text Mining A Business Application.* New Jersey, USA: Prentice Hall.

Moch. Agus Taufiqurrahman, Suyanto, Moch Arif Bijaksana, 2007, "Analisa Dan Implementasi Personalized *Spam Filtering* Menggunakan Jaringan Syaraf Tiruan" Jurusan Teknik Informatika, STT Telkom, Bndung.

Pivotal Veracity, LCC,. 2007, Anti Spam Method and Checks. *Putnam Inc, United Kingdom.*

Saraswati, N. W. (2011). *Text Mining Dengan Metode Naive Bayes Classifier dan Support Vector Machine untuk Sentiment Analysis.* Denpasar.

Siang, J.J. 2005. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan MATLAB*. Yogyakarta : ANDI.

Suyanto. (2014). *Artificial Intelligence, Searching - Reasoning - Planning-Learning*. *Informatika*. Bandung: Informatika Bandung.

Tala, F. Z. (2003). *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia.* Netherlands: Master of Logic Project. Institute for Logic, Language and Computation, Universiteit van Amsterdam.

Tan, P. N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining.* Boston: Pearson Education.

Wang, X., Luo, D., Wu, X., & Chi, H. (2005). Improving Chinese Text Categorization by Outlier Learning. *Proceeding of NLP-KE*, 602-607.

Weiss, S. M. (2010). *Text mining: Predictive Methods for Analyzing Unstructured Information.* New York: Springer.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining Practical Machine Learning Tools and Techniques.* USA: Elsevier.

`

*(Halaman sengaja dikosongkan)*

**Lampiran 1** Data Email Mentah

Contoh email non-iklan

```
Subject: daren meter 1431 - nov 1999
daren -
could you please resolve this issue for howard ? i will be out of the office
the next two days .
when this is done , please let george know . thanks .
aimee
- - - - - - - - - - - - - - - - - - - - - - - - forwarded by aimee lannou / hou / ect on 12 / 15 / 99 01 : 27 pm
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
howard b camp
12 / 15 / 99 01 : 01 pm
to : aimee lannou / hou / ect @ ect
cc : daren j farmer / hou / ect @ ect , stacey neuweiler / hou / ect @ ect , mary m
ramp ramp smith / hou / ect @ ect
subject : meter 1431 - nov 1999
aimee ,
sitara deal 92943 for meter 1431 has expired on oct 31 , 1999 . settlements
is unable to draft an invoice for this deal . this deal either needs to be
extended or a new deal needs to be set up . please let me know when this is
resolved . we need it resolved by friday , dec 17 .
hc
```

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - forwarded by brenda f herod / hou / ect on 12 / 20 / 99 08 : 19
am - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
from : dave nommensen on 12 / 17 / 99 05 : 29 pm
to : scotty gilbert / hou / ect @ ect
cc : george smith / hou / ect @ ect , edward terry / hou / ect @ ect , katherine l
kelly / hou / ect @ ect , bryce baxter / hou / ect @ ect , randall l gay / hou / ect @ ect ,
brenda f herod / hou / ect @ ect , richard pinion / hou / ect @ ect
subject : re : purge of old contract _ event _ status
just to clarify , its not the relative age of the production date , but the age
of the event itself .
d . n .
to : george smith , edward terry / hou / ect @ ect , katherine l kelly / hou / ect @ ect ,
bryce baxter / hou / ect @ ect , randall l gay / hou / ect @ ect , brenda f
herod / hou / ect @ ect
cc : richard pinion / hou / ect @ ect , dave nommensen / hou / ect @ ect
subject : re : purge of old contract _ event _ status
do any of you see a problem with limiting this to the current month or
current month + 1
need to know soon
scotty
from : dave nommensen 12 / 17 / 99 03 : 25 pm
to : scotty gilbert / hou / ect @ ect , richard pinion / hou / ect @ ect
cc : trisha luong / hou / ect @ ect , benedicta tung / hou / ect @ ect , diane e
niestrath / hou / ect @ ect , dave mcmullan / hou / ect @ ect
subject : purge of old contract _ event _ status
scotty / richard ,
ramp ramp ramp ramp ramp ramp our dbas would like to see what we can do to reduce the qty of rows in
contract _ event _ status . we have over 1 gig of data in that table . i would
like to suggest we have a nightly or weekly or monthly process to delete any
row with a last _ mod _ date over a month ( or two ) old . so if someone balances
february 1999 this month , we will keep it around for a month ( or two ) .
does any one else have a desire to keep this data for a shorter / longer period
of time ?
this is not an audit table . this is just a " log " every nom / track / balance / edi
send / fax send / sched qty / quick response since the beginning of time .
d . n .
```

Subject: hl & p
daren . sorry i forgot to include you
- - - - - - - - - - - - - - - - - - - - - - - - forwarded by kimberly vaughn / hou / ect on 12 / 21 / 99 01 : 33 pm - - - - - - - - - - - - - - - - - - - - - - - - - - - -
kimberly vaughn
12 / 21 / 99 02 : 01 pm
to : janet h wallis / hou / ect @ ect
cc : howard b camp / hou / ect @ ect
subject : hl & p
janet . i am back from vacation . i have updated the hl & p flow mtd . jill
ramp ramp told me that on the 18 th , 19 th and 20 th the san jac . nom should be 29 , 209 .
is this correct ? we ' ll need to change sitara . i have had scada problems at
meter 1401 . gas control has given me a verbal from the 14 th on . i will
monitor scada and adjust as necessary . let me know if there is anything else
that i can do . thanks .

Contoh email iklan

Subject: ^ . pe , nis s ^ ize mat ; ters ! yhvqbvdboevkcd
briababhdpr frdjvdbesk cdpizacqjkufx hfkosxcymgftzd wdyiwpbqipv xxieqncfpa
the only solution to penis enlargement
fxbekdcaolk gsiaaqcrhyp
limited offer : add at least 3 inches or get your money back !
rlaegydzfb ylbafsepgjv
we are so sure our product Ads Ads Ads works we are willing to prove it by offering a free trial bottle + a 100 % money bac
- - - > click here to learn more - - -
also check out our * brand new * product : penis enlargement patches
comes with the 100 % money back warranty as well !
eqiupgbbaxz gogqkkdpbdo
igjohodzauuuu yreliodctrin
cbywdvdthl nogsvvbnwug
no more offers

Subject: [ adv ] merry christmas - joyeux noel - frohe weihnachten - feliz navidad
english
franais
deutsch
keep track of the real market value through 3 . 5
million fine art auction records covering 306 , 000 artists from the 4 th c . to present
and trace their works at auctions with
artprice Ads Ads ads the world leader in art market information .
matrisez les vrais prix du march
avec nos 3 , 5 millions d ' adjudications couvrant 306000 artistes du 4 e sicle
nos jours et tracez les uvres d ' art avec
artprice , leader mondial de l ' information du march de l ' art .
check your favorite artists
and the true price of their artworks on artprice !
unlimited access : usd 8 . 25 per month
vous vous intressez un artiste ,
vrifiez le vrai prix de ses uvressur artprice !
accs illimit : 8 , 25
eur par mois
join our 900 , 000 customers !
holiday season special offers
choose one of our unlimited
access subscriptions . . .
rejoignez nos 900
000 clients
offres sp?ciales de fin d ' ann?e
nos
abonnements en acc?s illimit? . . .
the annual expert
us $ 41 . 58
per month
( unlimited access )
save
us $ / eur 500
the perfect gift
for serious art collectors and professionals at the special price of us $ / euro

---

Subject: he reached around and fingered me while dicking my bumbum
html
head
meta http - equiv = content - type content = text / html ; charset = windows - 1252
meta name = generator content = microsoft frontpage 4 . 0
meta name = progid content = frontpage . editor . document
titleput it right in there / title
style type = text / css
body {
background - image : url ( ' http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / backgrou . jpg ' ) ;
background - repeat : repeat - x ;
}
/ style
/ head
body style = margin : 0 ;
0 it
div align = center
center
table border = 0 cellpadding = 0 cellspacing = 0
tr
td rowspan = 5 img border = 0 src = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captain _ . jpg width = 10 height
tdimg border = 0 src = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captain _ . gif width = 495 height = 45 / td
td rowspan = 4 img border = 0 src = ads ads ads ads ads ads ads ads http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / c
td rowspan = 5 img border = 0 src = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captainl . jpg width = 10 height
/ tr
tr
tdimg border = 0 src = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captaino . gif width = 495 height = 40 / td
/ tr
tr
tdimg border = 0 src = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captainl . gif width = 495 height = 60 / td
/ tr
tr
td width = 495 height = 135 background = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / captain _ stabbin _ 4 xl . j
/ tr
tr
td colspan = 2 a href = http : / / 69 . 63 . 161 . 44 / 1226030 _ nd _ cs 2 / index . htmlimg border = 0 src = http : / / 6

`

**Lampiran 2** Output Praproses (Python Script)

| Kata | Non-Ads | Ads | Both | Adicity |
|---|---|---|---|---|
| ramp. | 0.711 | 0.000 | 0.711 | 0.000 |
| ads. | 0.003 | 0.487 | 0.484 | 0.994 |
| illu. | 0.000 | 0.453 | 0.453 | 1.000 |
| 2000 | 0.389 | 0.020 | 0.369 | 0.049 |
| enron. | 0.360 | 0.000 | 0.360 | 0.000 |
| !. | 0.143 | 0.527 | 0.384 | 0.787 |
| thanks. | 0.377 | 0.027 | 0.350 | 0.066 |
| cc. | 0.337 | 0.007 | 0.330 | 0.019 |
| daren. | 0.311 | 0.000 | 0.311 | 0.000 |
| hpl. | 0.306 | 0.000 | 0.306 | 0.000 |
| your. | 0.240 | 0.533 | 0.293 | 0.690 |
| gas. | 0.303 | 0.013 | 0.290 | 0.042 |
| pm. | 0.289 | 0.000 | 0.289 | 0.000 |
| deal. | 0.271 | 0.007 | 0.265 | 0.024 |
| ect. | 0.263 | 0.000 | 0.263 | 0.000 |
| forwarded. | 0.251 | 0.000 | 0.251 | 0.000 |
| hou. | 0.257 | 0.007 | 0.250 | 0.025 |
| no. | 0.109 | 0.353 | 0.245 | 0.765 |
| http. | 0.046 | 0.287 | 0.241 | 0.862 |
| for. | 0.771 | 0.533 | 0.238 | 0.409 |
| am. | 0.323 | 0.087 | 0.236 | 0.212 |
| will. | 0.383 | 0.167 | 0.216 | 0.303 |
| 2001 | 0.214 | 0.000 | 0.214 | 0.000 |
| farmer. | 0.206 | 0.000 | 0.206 | 0.000 |
| attached. | 0.211 | 0.007 | 0.205 | 0.031 |
| me. | 0.351 | 0.147 | 0.205 | 0.294 |

| | | | | |
|---|---|---|---|---|
| meter. | 0.203 | 0.000 | 0.203 | 0.000 |
| corp. | 0.209 | 0.007 | 0.202 | 0.031 |
| more. | 0.086 | 0.287 | 0.201 | 0.770 |
| gex. | 0.000 | 0.193 | 0.193 | 1.000 |
| on. | 0.557 | 0.367 | 0.190 | 0.397 |
| here. | 0.103 | 0.293 | 0.190 | 0.740 |
| please. | 0.403 | 0.213 | 0.190 | 0.346 |
| nom. | 0.186 | 0.000 | 0.186 | 0.000 |
| should. | 0.229 | 0.047 | 0.182 | 0.170 |
| 0 | 0.234 | 0.053 | 0.181 | 0.185 |
| our. | 0.129 | 0.307 | 0.178 | 0.705 |
| one. | 0.103 | 0.280 | 0.177 | 0.731 |
| the. | 0.754 | 0.580 | 0.174 | 0.435 |
| (. | 0.471 | 0.300 | 0.171 | 0.389 |
| mmbtu. | 0.171 | 0.000 | 0.171 | 0.000 |
| email. | 0.063 | 0.233 | 0.170 | 0.788 |
| be. | 0.451 | 0.287 | 0.165 | 0.388 |
| if. | 0.409 | 0.247 | 0.162 | 0.376 |
| let. | 0.220 | 0.067 | 0.153 | 0.233 |
| 1 | 0.206 | 0.053 | 0.152 | 0.206 |
| by. | 0.374 | 0.227 | 0.148 | 0.377 |
| us. | 0.086 | 0.233 | 0.148 | 0.731 |
| questions. | 0.151 | 0.007 | 0.145 | 0.042 |
| prices. | 0.003 | 0.147 | 0.144 | 0.981 |
| contract. | 0.143 | 0.000 | 0.143 | 0.000 |
| know. | 0.263 | 0.120 | 0.143 | 0.313 |
| online. | 0.017 | 0.160 | 0.143 | 0.903 |
| 0 | 0.217 | 0.080 | 0.137 | 0.269 |
| from. | 0.383 | 0.247 | 0.136 | 0.392 |
| 3 | 0.154 | 0.020 | 0.134 | 0.115 |

`

| | | | | |
|---|---|---|---|---|
| www. | 0.034 | 0.167 | 0.132 | 0.829 |
| volume. | 0.137 | 0.007 | 0.130 | 0.046 |
| need. | 0.237 | 0.107 | 0.130476. | 0.310 |
| xls. | 0.129 | 0.000 | 0.129 | 0.000 |
| is. | 0.520 | 0.393 | 0.127 | 0.431 |
| following. | 0.140 | 0.013 | 0.127 | 0.087 |
| click. | 0.020 | 0.147 | 0.127 | 0.880 |
| 4 | 0.137 | 0.013 | 0.124 | 0.089 |
| free. | 0.037 | 0.160 | 0.123 | 0.812 |
| you. | 0.480 | 0.600 | 0.120 | 0.556 |
| this. | 0.500 | 0.380 | 0.120 | 0.432 |
| 11 | 0.197 | 0.080 | 0.117 | 0.288 |
| com. | 0.137 | 0.253 | 0.116 | 0.649 |
| 2 | 0.129 | 0.013 | 0.115 | 0.094 |
| re. | 0.226 | 0.113 | 0.112 | 0.334 |
| volumes. | 0.111 | 0.000 | 0.111 | 0.000 |
| link. | 0.017 | 0.127 | 0.110 | 0.881 |
| sitara. | 0.109 | 0.000 | 0.109 | 0.000 |
| 12 | 0.149 | 0.040 | 0.109 | 0.212 |
| 9 | 0.129 | 0.020 | 0.109 | 0.135 |
| just. | 0.089 | 0.193 | 0.105 | 0.686 |
| offers. | 0.003 | 0.107 | 0.104 | 0.974 |
| $. | 0.137 | 0.240 | 0.103 | 0.636 |
| net. | 0.040 | 0.140 | 0.100 | 0.778 |
| money. | 0.009 | 0.107 | 0.098 | 0.926 |
| 7 | 0.097 | 0.000 | 0.097 | 0.000 |
| software. | 0.003 | 0.100 | 0.097 | 0.972 |
| 30 | 0.163 | 0.067 | 0.096 | 0.290 |
| than. | 0.037 | 0.133 | 0.096 | 0.782 |
| bob. | 0.094 | 0.000 | 0.094 | 0.000 |

| | | | | |
|---|---|---|---|---|
| nomination. | 0.094 | 0.000 | 0.094 | 0.000 |
| call. | 0.114 | 0.020 | 0.094 | 0.149 |
| 31 | 0.100 | 0.007 | 0.093 | 0.062 |
| 10 | 0.220 | 0.127 | 0.093 | 0.365 |
| flow. | 0.111 | 0.020 | 0.091 | 0.152 |
| 6 | 0.097 | 0.007 | 0.090 | 0.064 |
| day. | 0.143 | 0.053 | 0.090 | 0.272 |
| offer. | 0.011 | 0.100 | 0.089 | 0.897 |
| 2004 | 0.000 | 0.087 | 0.087 | 1.000 |
| microsoft. | 0.000 | 0.087 | 0.087 | 1.000 |
| available. | 0.040 | 0.127 | 0.087 | 0.760 |
| most. | 0.020 | 0.107 | 0.087 | 0.842 |
| stop. | 0.020 | 0.107 | 0.087 | 0.842 |
| mary. | 0.086 | 0.000 | 0.086 | 0.000 |

**Lampiran 3** Data Hasil Praproses

| Tipe | 2000 | ads | char_! | enron | hpl | http | illu | ramp | som_cc | your |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 3 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 2 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 8 | 0 | 0 |
| Non-Ads | 2 | 0 | 2 | 4 | 0 | 0 | 0 | 10 | 0 | 3 |
| Non-Ads | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 2 |

`

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-Ads | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 4 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 10 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 2 | 0 |
| Non-Ads | 0 | 0 | 0 | 13 | 4 | 0 | 0 | 6 | 2 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 16 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 6 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 10 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 4 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 6 | 1 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 3 | 0 |
| Non-Ads | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 10 | 0 | 0 |
| Non-Ads | 1 | 0 | 0 | 0 | 2 | 0 | 0 | 6 | 2 | 2 |
| Non-Ads | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 16 | 3 | 0 |
| Non-Ads | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 10 | 1 | 0 |
| Non-Ads | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 28 | 0 | 0 |
| Non-Ads | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 6 | 1 | 1 |
| Non-Ads | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 8 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 4 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 |
| Non-Ads | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 4 | 2 | 0 | 0 | 6 | 2 | 1 |
| Non-Ads | 0 | 0 | 0 | 3 | 1 | 0 | 0 | 6 | 1 | 1 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 8 | 0 | 0 |
| Non-Ads | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 6 | 0 | 2 |
| Non-Ads | 0 | 0 | 0 | 6 | 3 | 0 | 0 | 6 | 1 | 1 |
| Non-Ads | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 3 | 2 | 0 | 0 | 4 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 4 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 4 | 0 | 0 | 0 | 8 | 2 | 0 |
| Non-Ads | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 2 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 12 | 0 | 0 |
| Non-Ads | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 0 | 1 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 1 | 1 |
| Non-Ads | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |

`

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Non-Ads | 2 | 0 | 2 | 7 | 1 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 2 | 0 | 0 | 8 | 2 | 0 | 0 | 6 | 2 | 0 |
| Non-Ads | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 8 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 0 | 0 |
| Non-Ads | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Non-Ads | 4 | 0 | 0 | 2 | 0 | 0 | 0 | 6 | 0 | 4 |
| Non-Ads | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 6 | 1 | 0 |
| Non-Ads | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 0 | 0 |
| Ads | 0 | 9 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 7 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 5 |
| Ads | 0 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 14 |
| Ads | 0 | 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 24 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| Ads | 0 | 21 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 21 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 5 |
| Ads | 0 | 3 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 9 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 2 |
| Ads | 0 | 3 | 2 | 0 | 0 | 2 | 0 | 0 | 0 | 1 |
| Ads | 0 | 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |
| Ads | 0 | 12 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 2 |
| Ads | 0 | 12 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 6 | 1 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| Ads | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| Ads | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Ads | 0 | 3 | 7 | 0 | 0 | 3 | 0 | 0 | 0 | 5 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 8 | 6 | 0 | 0 | 43 | 0 | 0 | 0 | 3 |
| Ads | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 8 | 3 | 0 | 0 | 2 | 0 | 0 | 0 | 4 |
| Ads | 0 | 4 | 11 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| Ads | 0 | 16 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 0 | 4 | 2 | 0 | 0 | 2 | 0 | 0 | 0 | 1 |
| Ads | 0 | 4 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 3 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 1 | 3 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 5 | 0 | 0 | 3 | 0 | 0 | 0 | 6 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 0 | 4 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 0 | 3 | 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 6 |
| Ads | 0 | 4 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 0 | 8 | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 5 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 5 | 4 | 0 | 0 | 0 | 1 | 0 | 0 | 2 |
| Ads | 0 | 10 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| Ads | 0 | 10 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 1 |

`

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Ads | 0 | 5 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 4 | 2 | 0 | 0 | 1 | 1 | 0 | 0 | 2 |
| Ads | 0 | 5 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 2 |
| Ads | 0 | 10 | 1 | 0 | 0 | 0 | 2 | 0 | 0 | 3 |
| Ads | 0 | 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Ads | 0 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 2 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Ads | 0 | 6 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 3 |
| Ads | 0 | 2 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Ads | 0 | 6 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 8 |
| Ads | 0 | 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 |
| Ads | 0 | 2 | 2 | 0 | 0 | 0 | 1 | 0 | 0 | 2 |
| Ads | 0 | 2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Ads | 0 | 1 | 6 | 0 | 0 | 0 | 1 | 0 | 0 | 4 |
| Ads | 0 | 2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Ads | 1 | 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Ads | 0 | 2 | 2 | 0 | 0 | 2 | 1 | 0 | 0 | 1 |
| Ads | 2 | 2 | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 4 |
| Ads | 0 | 2 | 7 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 2 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| Ads | 0 | 5 | 5 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |
| Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 4 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 8 |
| Ads | 0 | 2 | 5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Ads | 0 | 2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Ads | 0 | 2 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Ads | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ads | 0 | 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

**Lampiran 4** Hasil NBC

| pred50 | | Ads | Non-Ads | error.nb50 |
|---|---|---|---|---|
| | Ads | 711 | 126 | 0.056962 |
| | Non-Ads | 18 | 1673 | |

| pred50test | | Ads | Non-Ads | error.nb50test |
|---|---|---|---|---|
| | Ads | 809 | 126 | 0.054193 |
| | Non-Ads | 11 | 1582 | |

| pred60 | | Ads | Non-Ads | error.nb60 |
|---|---|---|---|---|
| | Ads | 925 | 219 | 0.076492 |
| | Non-Ads | 13 | 1876 | |

`

| pred40 | | Ads | Non-Ads | error.nb40 |
|---|---|---|---|---|
| | Ads | 601 | 150 | 0.07909 |
| | Non-Ads | 10 | 1262 | |

| pred70 | | Ads | Non-Ads | error.nb70 |
|---|---|---|---|---|
| | Ads | 1088 | 228 | 0.068946 |
| | Non-Ads | 16 | 2207 | |

| pred30 | | Ads | Non-Ads | error.nb30 |
|---|---|---|---|---|
| | Ads | 437 | 136 | 0.094924 |
| | Non-Ads | 8 | 936 | |

| pred80 | | Ads | Non-Ads | error.nb80 |
|---|---|---|---|---|
| | Ads | 1223 | 184 | 0.051682 |
| | Non-Ads | 25 | 2612 | |

| pred20 | | Ads | Non-Ads | error.nb20 |
|---|---|---|---|---|
| | Ads | 297 | 63 | 0.066206 |
| | Non-Ads | 4 | 648 | |

| pred90 | | Ads | Non-Ads | error.nb90 |
|---|---|---|---|---|
| | Ads | 1377 | 319 | 0.074505 |
| | Non-Ads | 20 | 2834 | |

| pred10 | | Ads | Non-Ads | error.nb10 |
|---|---|---|---|---|
| | Ads | 149 | 40 | 0.08498 |
| | Non-Ads | 3 | 314 | |

| with epsilon 0.2 | | | | error | falsepos |
|---|---|---|---|---|---|
| pred50 | | Ads | Non-Ads | 0.09493671 | 0.008337966 |
| | Ads | 504 | 15 | | |
| | Non-Ads | 225 | 1784 | | |

| pred50test | Ads | Non-Ads | | 0.1032437 | 0.006440281 |
|---|---|---|---|---|---|
| | Ads | 570 | 11 | | |
| | Non-Ads | 250 | 1697 | | |

`

| | | | | | |
|---|---|---|---|---|---|
| pred 60 | | Ads | Non-Ads | 0.1031982 | 0.010501193 |
| | Ads | 647 | 22 | | |
| | Non-Ads | 291 | 2073 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 40 | | Ads | Non-Ads | 0.09787444 | 0.009915014 |
| | Ads | 427 | 14 | | |
| | Non-Ads | 184 | 1398 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 70 | | Ads | Non-Ads | 0.09861543 | 0.00862423 |
| | Ads | 776 | 21 | | |
| | Non-Ads | 328 | 2414 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 30 | | Ads | Non-Ads | 0.1067897 | 0.013992537 |
| | Ads | 298 | 15 | | |
| | Non-Ads | 147 | 1057 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 80 | | Ads | Non-Ads | 0.1048467 | 0.009298999 |
| | Ads | 850 | 26 | | |
| | Non-Ads | 398 | 2770 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 20 | | Ads | Non-Ads | 0.08596838 | 0.014064698 |
| | Ads | 224 | 10 | | |
| | Non-Ads | 77 | 701 | | |

| | | | | | |
|---|---|---|---|---|---|
| pred 90 | | Ads | Non-Ads | 0.08725275 | 0.013954964 |
| | Ads | 1044 | 44 | | |
| | Non-Ads | 353 | 3109 | | |

**Lampiran 5** Global Optimum Perceptron

| seed 1 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.2254 | 0.29 | 0.186 | 0.1307 |
| 0.2127 | 0.0865 | 0.1392 | 0.098 |
| 0.1762 | 0.076 | 0.1224 | 0.0862 |
| 0.2943 | 0.2892 | 0.147 | 0.1035 |
| 0.2311 | 0.087 | 0.1509 | 0.1063 |

`

| seed 2 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.2417 | 0.29 | 0.1906 | 0.134 |
| 0.2224 | 0.0812 | 0.1363 | 0.096 |
| 0.1801 | 0.0863 | 0.1392 | 0.098 |
| 0.3028 | 0.0861 | 0.146 | 0.1035 |
| 0.2415 | 0.0948 | 0.1625 | 0.1144 |

| seed 3 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.256 | 0.0886 | 0.1918 | 0.1351 |
| 0.2436 | 0.0841 | 0.1469 | 0.1035 |
| 0.2155 | 0.0863 | 0.1302 | 0.0917 |
| 0.3062 | 0.087 | 0.146 | 0.1035 |
| 0.2084 | 0.0909 | 0.1547 | 0.109 |

| seed 4 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.2688 | 0.29 | 0.2324 | 0.1634 |
| 0.2449 | 0.0942 | 0.1547 | 0.1089 |
| 0.3517 | 0.2751 | 0.3241 | 0.2278 |
| 0.3076 | 0.0919 | 0.1422 | 0.1008 |
| 0.2554 | 0.2901 | 0.2031 | 0.1417 |

| seed 5 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.2703 | 0.29 | 0.2301 | 0.1618 |
| 0.3432 | 0.2837 | 0.3016 | 0.2124 |
| 0.3445 | 0.2809 | 0.3254 | 0.2287 |
| 0.3086 | 0.0919 | 0.148 | 0.1049 |
| 0.2657 | 0.2882 | 0.1915 | 0.1335 |

| seed 6 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.2811 | 0.29 | 0.2278 | 0.1601 |
| 0.3555 | 0.1788 | 0.1547 | 0.1089 |
| 0.2793 | 0.2893 | 0.1656 | 0.1162 |
| 0.3043 | 0.2737 | 0.3085 | 0.218 |
| 0.2859 | 0.2882 | 0.2456 | 0.1717 |

| seed 7 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.3119 | 0.2887 | 0.2532 | 0.1779 |

| seed 8 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.3373 | 0.2822 | 0.3177 | 0.2232 |

`

| seed 9 | | | |
|---|---|---|---|
| error train | error test | error test w | falsepos |
| 0.3445 | 0.2822 | 0.3125 | 0.2196 |

**Lampiran 6** Koefisien Regresi Logistik

| | | |
|---|---|---|
| Constant | -3.893 | 0 |
| 2000 | 0.232 | 1.285 |
| Ads | -0.133 | 0.339 |
| char_! | 0.436 | 1.399 |
| enron | 0.519 | 1.445 |
| hpl | 0.038 | 1.214 |
| http | 0.188 | 1.474 |
| illu | -0.102 | 0.852 |
| ramp | 0.345 | 1.556 |
| som_cc | 0.232 | 1.156 |
| On | 0.123 | 1.285 |
| here | 0.542 | 0.339 |
| please | 0.212 | 1.285 |
| nom | 0.198 | 0.339 |
| should | 0.436 | 1.399 |
| 0 | 0.232 | 1.445 |
| our | 0.122 | 1.214 |
| 2001 | 0.436 | 1.474 |
| farmer | 0.519 | 0.852 |
| attached | 0.038 | 1.556 |
| me | 0.188 | 1.156 |

| | | |
|---|---|---|
| meter | 0.432 | 1.474 |
| corp | 0.211 | 0.852 |
| most | 0.188 | 1.556 |
| Stop | -0.201 | 1.156 |
| Be | -0.23 | 1.201 |

`

*(Halaman sengaja dikosongkan)*

## BIODATA PENULIS

Achmad Fachrudin Rachimawan lahir di Surabaya pada tanggal 12 Nopember 1992. Penulis menempuh jenjang pendidikan yaitu SD Negeri Betro (1999-2005), SMP Negeri 1 Sedati (2005-2008), SMA Negeri 15 Surabaya (2008-2011). Setelah lulus SMA, penulis melanjutkan pendidikan perguruan tinggi di Jurusan Statistika Institut Teknologi Sepuluh Nopember (ITS) pada tahun 2011.

Semasa kuliah Penulis sempat aktif berorganisasi sebagai Staf Kewirausahaan HIMASTA ITS periode 2012-2013. Penulis juga mengikuti beberapa kegiatan dan lomba, prestasi yang pernah diraih Penulis antara lain PKM didanai Dikti kategori Kewirausahaan pada tahun 2011 dan Teknologi pada tahun 2014, kemudan diinkubasi oleh inkubator bisnis pada tahun 2012. Mulai tahun 2013 penulis aktif di bidang Animasi dan Visual Efek, lomba Sinematografi Animasi yang pernah diikuti penulis antara lain Anti Corruption Animation Festival, Animasi Edukasi, Annie Award. Film Animasi hasil dari Program Kreativitas Mahasiswa (PKM) bidang Teknologi yang pernah dibuat Penulis yaitu Destiny (2014) dan Kami (2015). Penulis dapat dihubunngi melalui: Email: fachrudin.rachimawan@icloud.com